Final Report

Title: Information-Based Approach to Unsupervised Machine Learning

AFOSR/AOARD Reference Number: FA2386-11-1-4059

AFOSR/AOARD Program Manager: Hiroshi Motoda, Ph.D.

Period of Performance: 24 months from 20 June, 2011

Submission Date: 19 June, 2013

PI: Masashi Sugiyama, Tokyo Institute of Technology

Report Documentation Page

Form Approved
OMB No. 0704-0188

1. REPORT DATE **19 JUN 2013**	2. REPORT TYPE **Final**	3. DATES COVERED **20-06-2011 to 19-06-2013**

4. TITLE AND SUBTITLE
Information-Based Approach to Unsupervised Machine Learning

5a. CONTRACT NUMBER
FA23861114059

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Masashi Sugiyama

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Department of Computer Science ,Tokyo Institute of Technology
,2-12-1-W8-74, O-okayama, ,Meguro-ku, Tokyo 152-8552,NA,NA

8. PERFORMING ORGANIZATION REPORT NUMBER
N/A

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
AOARD, UNIT 45002, APO, AP, 96338-5002

10. SPONSOR/MONITOR’S ACRONYM(S)
AOARD

11. SPONSOR/MONITOR’S REPORT NUMBER(S)
AOARD-114059

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

Unsupervised machine learning methods such as clustering and change detection are indispensable to various real-world data processing tasks. However, due to its vague formulation, studies of unsupervised learning tend to be ad-hoc, and thus development of unsupervised learning methods is still far behind supervised learning. The project aims at overcoming this difficulty by providing a systematic approach to unsupervised learning based on information measures. The PI and his group developed various information-based machine learning algorithms, including clustering, independence testing, object matching class-imbalance adaptation, change detection, and canonical dependency analysis. Furthermore, they explored fundamental data processing paradigms for further improving accuracy and robustness of information estimators in high dimensional problems. Through this project, they advanced the field of unsupervised learning by providing a novel systematic approach based on information measures.

15. SUBJECT TERMS
Brain Science and Engineering; Cognitive Neuroscience; Human-Computer Interface; Intelligent Systems

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT **unclassified**	b. ABSTRACT **unclassified**	c. THIS PAGE **unclassified**	**Same as Report (SAR)**	**226**	

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std Z39-18

1 Objectives

Supervised machine learning can be formulated in a mathematically rigorous way as the problem of inferring an underlying functional relation behind data. However, unsupervised machine learning is often defined in an ambiguous way—extracting some useful knowledge hidden in the data without explicit guidance. Nevertheless, unsupervised learning methods such as clustering and change detection are indispensable to various real-world applications. However, due to its vague formulation, studies of unsupervised learning tend to be ad-hoc, and thus development of unsupervised learning methods is still far behind supervised learning. The aim of this project is to overcome this difficulty by providing a systematic approach to a class of ill-defined unsupervised learning problems based on *information measures*.

Mutual information (MI) is a standard information measure that has been extensively explored in the field of information theory. However, MI is hard to approximate from data samples and it is not robust against outliers. In our project, we consider an alternative information measure called *squared-loss mutual information* (SMI), which is more robust against outliers by definition. To develop a family of machine learning algorithms based on SMI, we utilize its robust and computationally efficient approximator called *least-squares mutual information* (LSMI), which is one of the major deliverables of my previous project, "*A Density-Ratio Approach to Machine Learning*", supported by AFOSR/AOARD (AOARD-09-4071). The usefulness of the proposed approach is demonstrated through experiments.

2 Status of effort

My project consists of two subjects: (A) Development of information-based machine learning algorithms and (B) Improvement of information estimators for further advances.

For the subject (A), we have actively explored various unsupervised machine learning topics and developed novel information-based algorithms, including clustering, independence testing, object matching, class-imbalance adaptation, change detection, and canonical dependency analysis. We further developed methods of supervised dimension reduction, probabilistic classification, and non-stationarity adaptation in the same framework.

For the subject (B), we explored novel paradigms for further improving the accuracy and robustness of information estimators, including information estimation with dimension reduction for coping with high dimensionality, information estimation with a relative-divergence and a difference-divergence for enhancing robustness against outliers, and a unified framework of information estimation for better understanding mutual relation among different information measures.

3 Abstract

We developed various information-based machine learning algorithms:

- Object matching: Given two sets of unpaired objects (such as speech signals from two different subjects, a set of photos and a photo frame, and images taken from different modalities), we pair them by maximizing their mutual dependency (Publication 4).
- Clustering: Given input-only samples, we determine their cluster labels by finding the most dependent label assignments on the original input samples (Publications 5 and 14).
- Canonical dependency analysis: Given two sets of paired samples, we find the projections to maximize their dependencies (Publication 20).
- Statistical testing: Given two sets of samples, we decide whether they are drawn from the same probability distribution (Publications 13 and 17). Similarly, given paired samples, we decide whether they are independence (Publication 11).
- Class-prior change adaptation: Given labeled training data and unlabeled test data having different class balances, we estimate the class-balance of unlabeled test data by matching the distribution of unlabeled test data with the class-wise mixture of training data (Publication 9).
- Change-detection in time-series: Given time-series, we detect change points at which properties of time-series switch by comparing the probability distributions of current and past data (Publication 19).
- Given labeled training data and unlabeled test data having different input distributions, we perform distribution-adaptive learning for reinforcement learning (Publication 15) and probabilistic classification (Publication 16).
- Supervised dimension reduction: Given input-output paired data, we reduce the dimensionality of input data by maximizing the dependency (Publication 7 and 24).
- Computationally efficient probabilistic classification: Given labeled data, we estimate the posterior probability of labels given an input pattern in a computationally efficient way (Publication 12).

We also investigated various properties of information estimators for further development:

- Elucidation of statistical and numerical properties of a least-squares kernel-based information estimator (Publication 18 and 25).
- Information estimation with dimension reduction for coping with high dimensionality (Publication 6).

- Information estimation with a relative-divergence and a difference-divergence for enhancing robustness against outliers (Publications 8, 26, and 10).
- A unified framework of information estimation for better understanding mutual relation among different information measures (Publication 21).
- Relation to a kernel-based independence measure (Publication 22).

Finally, we published monographs and review articles related to the current project:

- Monograph on density-ratio estimation (Publication 1).
- Monograph on non-stationarity adaptation (Publication 2).
- Review article on non-stationarity adaptation (Publication 3).
- Review article on information-based learning (Publication 23).

4 Personnel Supported

The research activity of the following people was supported.

- Masashi Sugiyama (Tokyo Institute of Technology),
- Makoto Yamada (Tokyo Institute of Technology),
- Gang Niu (Tokyo Institute of Technology),
- Marthinus Christoffel du Plessis (Tokyo Institute of Technology).

5 Publications

During the 24 months, the following papers were published. The papers indicated by '*' were attached to this report, and all the publications are available from

"`http://sugiyama-www.cs.titech.ac.jp/~sugi/publications.html`".

Books and Articles

1. Sugiyama, M., Suzuki, T., & Kanamori, T. Density Ratio Estimation in Machine Learning, 344 pages, Cambridge University Press, Cambridge, UK, 2012.
2. Sugiyama, M. & Kawanabe, M. Machine Learning in Non-Stationary Environments: Introduction to Covariate Shift Adaptation, 308 pages, MIT Press, Cambridge, MA, USA, 2012.
3. * Sugiyama, M. Learning under non-stationarity: covariate shift adaptation by importance weighting. In Handbook of Computational Statistics: Concepts and Methods, 2nd edition, Chapter 31, pp.927-952, Springer, Berlin, Germany, 2012.

Conference Papers

4. * Yamada, M. & Sugiyama, M. Cross-domain object matching with model selection. In Proceedings of Fourteenth International Conference on Artificial Intelligence and Statistics (AISTATS2011), vol.15, pp.807-815, Fort Lauderdale, Florida, USA, Apr. 11-13, 2011.

5. * Sugiyama, M., Yamada, M., Kimura, M., & Hachiya, H. On information-maximization clustering: Tuning parameter selection and analytic solution. In Proceedings of 28th International Conference on Machine Learning (ICML2011), pp.65-72, Bellevue, Washington, USA, Jun. 28-Jul. 2, 2011.

6. Yamada, M. & Sugiyama, M. Direct density-ratio estimation with dimensionality reduction via hetero-distributional subspace analysis. In Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence (AAAI2011), pp.549-554, San Francisco, California, USA, Aug. 7-11, 2011.

7. Yamada, M., Niu, G., Takagi, J., & Sugiyama, M. Computationally efficient sufficient dimension reduction via squared-loss mutual information. In Proceedings of the Third Asian Conference on Machine Learning (ACML2011), vol.20, pp.247-262, Taoyuan, Taiwan, Nov. 13-15, 2011.

8. Yamada, M., Suzuki, T., Kanamori, T., Hachiya, H., & Sugiyama, M. Relative density-ratio estimation for robust distribution comparison. In Advances in Neural Information Processing Systems 24 (NIPS2011), pp.594-602, 2011.

9. * du Plessis, M. C. & Sugiyama, M. Semi-supervised learning of class balance under class-prior change by distribution matching. In Proceedings of 29th International Conference on Machine Learning (ICML2012), pp.823-830, Edinburgh, Scotland, Jun. 26-Jul. 1, 2012.

10. * Sugiyama, M., Suzuki, T., Kanamori, T., du Plessis, M. C., Liu, S., & Takeuchi, I. Density-difference estimation. In Advances in Neural Information Processing Systems 25 (NIPS2012), pp.692-700, 2012.

Journal Papers

11. Sugiyama, M. & Suzuki, T. Least-squares independence test. IEICE Transactions on Information and Systems, vol.E94-D, no.6, pp.1333-1336, 2011.

12. Yamada, M., Sugiyama, M., Wichern, G., & Simm, J. Improving the accuracy of least-squares probabilistic classifiers. IEICE Transactions on Information and Systems, vol.E94-D, no.6, pp.1337-1340, 2011.

13. * Sugiyama, M., Suzuki, T., Itoh, Y., Kanamori, T., & Kimura, M. Least-squares two-sample test. Neural Networks, vol.24, no.7, pp.735-751, 2011.

14. Kimura, M. & Sugiyama, M. Dependence-maximization clustering with least-squares mutual information, Journal of Advanced Computational Intelligence and Intelligent Informatics. vol.15, no.7, pp.800-805, 2011.

15. Hachiya, H., Peters, J., & Sugiyama, M. Reward weighted regression with sample reuse for direct policy search in reinforcement learning. Neural Computation, vol.23, no.11, pp.2798-2832, 2011.

16. Hachiya, H., Sugiyama, M. & Ueda, N. Importance-weighted least-squares probabilistic classifier for covariate shift adaptation with application to human activity recognition. Neurocomputing, vol.80, pp.93-101, 2012.

17. Kanamori, T., Suzuki, T., & Sugiyama, M. F-divergence estimation and two-sample homogeneity test under semiparametric density-ratio models. IEEE Transactions on Information Theory, vol.58, no.2, pp.708-720, 2012.

18. Kanamori, T., Suzuki, T., & Sugiyama, M. Statistical analysis of kernel-based least-squares density-ratio estimation. Machine Learning, vol.86, no.3, pp.335-367, 2012.

19. Kawahara, Y. & Sugiyama, M. Sequential change-point detection based on direct density-ratio estimation. Statistical Analysis and Data Mining, vol.5, no.2, pp.114-127, 2012.

20. Karasuyama, M. & Sugiyama, M. Canonical dependency analysis based on squared-loss mutual information. Neural Networks, vol.34, pp.46-55, 2012.

21. * Sugiyama, M., Suzuki, T., & Kanamori, T. Density ratio matching under the Bregman divergence: A unified framework of density ratio estimation. Annals of the Institute of Statistical Mathematics, vol.64, no.5, pp.1009-1044, 2012.

22. Sugiyama, M. & Yamada, M. On kernel parameter selection in Hilbert-Schmidt independence criterion. IEICE Transactions on Information and Systems, vol.E95-D, no.10, pp.2564-2567, 2012.

23. * Sugiyama, M. Machine learning with squared-loss mutual information. Entropy, vol.15, no.1, pp.80-112, 2013.

24. Suzuki, T. & Sugiyama, M. Sufficient dimension reduction via squared-loss mutual information estimation. Neural Computation, vol.25, no.3, pp.725-758, 2013.

25. Kanamori, T., Suzuki, T., & Sugiyama, M. Computational complexity of kernel-based density-ratio estimation: A condition number analysis. Machine Learning, vol.90, no.3, pp.431-460, 2013.

26. * Yamada, M., Suzuki, T., Kanamori, T., Hachiya, H., & Sugiyama, M. Relative density-ratio estimation for robust distribution comparison. Neural Computation, vol.25, no.5, pp.1324-1370, 2013.

6 Interactions

I had discussion with my program manager, Dr. Hiroshi Motoda, during the Third Asian Conference on Machine Learning in Taiwan on Nov. 13-15, 2011, and received detailed technical comments and suggestions for this project. On Nov. 20, 2012 at my office at Tokyo Institute of Technology, I had a meeting with Dr. Hiroshi Motoda and Lt Col Brian Sells and received comments and suggestions for further development.

Below is the list of my presentations related to the project.

1. Jul. 5, 2011: NEC Laboratories America, USA.
2. Jul. 21, 2011: ATR Computational Neuroscience Labs., Japan
3. Aug. 12, 2011: Yahoo! Research, USA.
4. Aug. 23, 2011: ERATO Project Meeting, Japan.
5. Sep. 15, 2011: Hokkaido University, Japan.
6. Oct. 21, 2011: SICE seminar, Japan.
7. Nov. 16, 2011: National Cheng Kung University, Taiwan.
8. Nov. 17, 2011: National Taiwan University, Taiwan.
9. Nov. 22, 2011: Symposium on Innovative Algorithms for e-Science, Japan.
10. Dec. 10, 2011: Empirical Inference Symposium, Germany.
11. Dec. 20, 2011: Toshiba Corporation, Japan.
12. Jan. 23, 2012: Mines ParisTech, France.
13. Jan. 24, 2012: Ecole Normale Superieure, France.
14. Jan. 25, 2012: INRIA Lille, France.
15. Jan. 27, 2012: FIRST Project Meeting, Japan.
16. Feb. 17, 2012: IPAB Seminar, Japan.
17. Apr. 25, 2012: Computational Science Simulation Symposium, Japan.
18. Jun. 11, 2012: Keio University, Japan.
19. Aug. 6, 2012: Workshop on Machine Learning and Applications to Biology, Japan.
20. Aug. 8, 2012: Hokkaido University, Japan.
21. Sep. 7, 2012: 21st Machine Learning Summer School, Japan.

22. Sep. 25, 2012: BBCI Summer School 2012, Germany.
23. Dec. 14, 2012: PRESTO Project Meeting, Japan.
24. Dec. 17, 2012: Toshiba Corporation, Japan.
25. Feb. 20, 2013: International Winter Workshop on Brain-Computer Interface, Korea.
26. Feb. 22, 2013: Seoul National University, Korea.
27. Feb. 26, 2013: NTT Communication Science Laboratories, Japan.
28. Mar. 6, 2013: Nagoya Institute of Technology, Japan.
29. Mar. 18, 2013: Aalto University, Finland.
30. Mar. 20, 2013: VALO Research and Trading, Finland.
31. Mar. 20, 2013: University of Helsinki, Finland.
32. Apr. 25, 2013: Omron Corporation, Japan.
33. May 21, 2013: JSAE-SICE Symposium, Japan.

7 Inventions

None.

8 Honors/Award

I received four awards related to the current project.

1. Jun. 19, 2012: IBISML Award Finalist, IEICE, Information-Based Induction Sciences and Machine Learning Technical Group.
2. Apr. 14, 2012: Funai Award, Funai Foundation for Information Technology.
3. Dec. 16, 2011: JNNS Best Paper Award, Japanese Neural Network Society.
4. Jun. 2, 2011: Nagao Special Researcher Award, Information Processing Society of Japan.

9 Archival Documentation

Selected papers (Publications 3, 4, 5, 9, 10, 13, 21, 23, and 26) are attached as archival documentation. All the publications listed in Section 5 are available from

“`http://sugiyama-www.cs.titech.ac.jp/~sugi/publications.html`”.

10 Software

Implementation of various machine learning algorithms (mostly in MATLAB) is available from my web page:

> “`http://sugiyama-www.cs.titech.ac.jp/~sugi/software/index.html`”.

In J. E. Gentle, W. Härdle, and Y. Mori (Eds.),
Handbook of Computational Statistics: Concepts and Methods,
2nd edition. Chapter 31, pp.927–952, Springer, Berlin, Germany, 2012.

Learning under Non-stationarity: Covariate Shift Adaptation by Importance Weighting

Masashi Sugiyama
Tokyo Institute of Technology
2-12-1 O-okayama, Meguro-ku, Tokyo 152-8552, Japan.
sugi@cs.titech.ac.jp http://sugiyama-www.cs.titech.ac.jp/˜sugi

Abstract

The goal of supervised learning is to estimate an underlying input-output function from its input-output training samples so that output values for unseen test input points can be predicted. A common assumption in supervised learning is that the training input points follow the *same* probability distribution as the test input points. However, this assumption is not satisfied, for example, when outside of the training region is extrapolated. The situation where the training and test input points follow *different* distributions while the conditional distribution of output values given input points is unchanged is called *covariate shift.* Since almost all existing learning methods assume that the training and test samples are drawn from the same distribution, their fundamental theoretical properties such as consistency or efficiency no longer hold under covariate shift. In this chapter, we review recently proposed techniques for covariate shift adaptation.

1 Introduction

The goal of supervised learning is to infer an unknown input-output dependency from training samples, by which output values for unseen test input points can be predicted. When developing a method of supervised learning, it is commonly assumed that the input points in the training set and the input points used for testing follow the *same* probability distribution (Wahba, 1990; Bishop, 1995; Vapnik, 1998; Duda et al., 2001; Hastie et al., 2001; Schölkopf & Smola, 2002). However, this common assumption is not fulfilled, for example, when outside of the training region is extrapolated or when training input points are designed by an active learning (a.k.a. experimental design) algorithm (Wiens, 2000; Kanamori & Shimodaira, 2003; Sugiyama, 2006; Kanamori, 2007; Sugiyama & Nakajima, 2009). Situations where training and test input points follow different probability distributions but the conditional distributions of output values given input points are unchanged are called *covariate shift* (Shimodaira, 2000). In this chapter, we review recently proposed techniques for alleviating for the influence of covariate shift.

Under covariate shift, standard learning techniques such as maximum likelihood estimation are biased. It was shown that the bias caused by covariate shift can be asymptotically canceled by weighting the loss function according to the *importance*—the ratio of test and training input densities (Shimodaira, 2000; Zadrozny, 2004; Sugiyama & Müller, 2005; Sugiyama et al., 2007; Quiñonero-Candela et al., 2009; Sugiyama & Kawanabe, 2011). Similarly, standard model selection criteria such as cross-validation (Stone, 1974; Wahba, 1990) or Akaike's information criterion (Akaike, 1974) lose their unbiasedness under covariate shift. It was shown that proper unbiasedness can also be recovered by modifying the methods based on importance weighting (Shimodaira, 2000; Zadrozny, 2004; Sugiyama & Müller, 2005; Sugiyama et al., 2007).

As explained above, the importance weight plays a central role in covariate shift adaptation. However, since the importance weight is unknown in practice, it should be estimated from data. A naive approach to this task is to first use kernel density estimation (Härdle et al., 2004) for obtaining estimators of the training and test input densities, and then taking the ratio of the estimated densities. However, division by estimated quantities can magnify the estimation error, so directly estimating the importance weight in a single-shot process would be more preferable. Following this idea, various methods for directly estimating the importance have been explored (Silverman, 1978; Ćwik & Mielniczuk, 1989; Qin, 1998; Cheng & Chu, 2004; Huang et al., 2007; Bickel et al., 2007; Sugiyama et al., 2008; Kanamori et al., 2009a). These direct estimation approaches have been demonstrated to be more accurate than the two-step density estimation approach.

Examples of successful real-world applications of covariate shift adaptation include brain-computer interface (Sugiyama et al., 2007), robot control (Hachiya et al., 2009; Akiyama et al., 2010; Hachiya et al., 2011), speaker identification (Yamada et al., 2010a), age prediction from face images (Ueki et al., 2011), wafer alignment in semiconductor exposure apparatus (Sugiyama & Nakajima, 2009), and natural language processing (Tsuboi et al., 2009).

The rest of this chapter is organized as follows. In Section 2, the problem of supervised learning under covariate shift is mathematically formulated. In Section 3, various learning methods under covariate shift are introduced. In Section 4, the issue of model selection under covariate shift is addressed. In Section 5, methods of importance estimation are reviewed. Finally, we conclude in Section 6.

A more extensive description of covariate shift adaptation techniques is available in Sugiyama and Kawanabe (2011).

2 Formulation of Supervised Learning under Covariate Shift

In this section, we formulate the supervised learning problem under covariate shift.

Let us consider the supervised learning problem of estimating an unknown input-

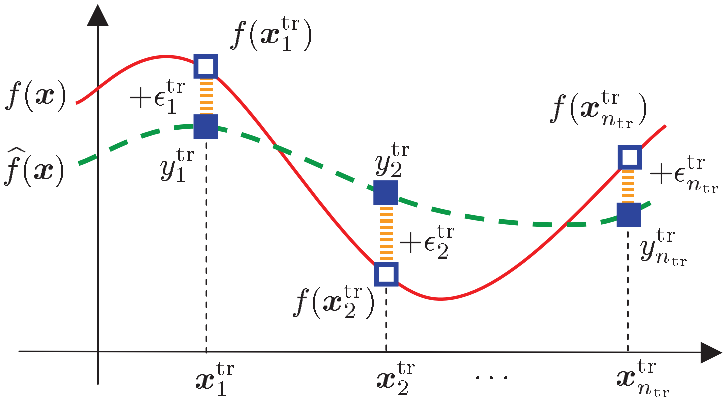

$f(\boldsymbol{x}_1^{\mathrm{tr}})$
$f(\boldsymbol{x})$
$+\epsilon_1^{\mathrm{tr}}$
$\widehat{f}(\boldsymbol{x})$
y_1^{tr}
$f(\boldsymbol{x}_{n_{\mathrm{tr}}}^{\mathrm{tr}})$
y_2^{tr}
$+\epsilon_{n_{\mathrm{tr}}}^{\mathrm{tr}}$
$+\epsilon_2^{\mathrm{tr}}$
$y_{n_{\mathrm{tr}}}^{\mathrm{tr}}$
$f(\boldsymbol{x}_2^{\mathrm{tr}})$
$\boldsymbol{x}_1^{\mathrm{tr}}$
$\boldsymbol{x}_2^{\mathrm{tr}}$
$\cdots$
$\boldsymbol{x}_{n_{\mathrm{tr}}}^{\mathrm{tr}}$

Figure 1: Framework of supervised learning.

output dependency from training samples. Let

$$\{(\boldsymbol{x}_i^{\mathrm{tr}}, y_i^{\mathrm{tr}}) \mid \boldsymbol{x}_i^{\mathrm{tr}} \in \mathcal{X} \subset \mathbb{R}^d, y_i^{\mathrm{tr}} \in \mathcal{Y} \subset \mathbb{R}\}_{i=1}^{n_{\mathrm{tr}}},$$

be the training samples. $\boldsymbol{x}_i^{\mathrm{tr}}$ is a training input point drawn from a probability distribution with density $p_{\mathrm{tr}}(\boldsymbol{x})$. y_i^{tr} is a training output value following a conditional probability distribution with conditional density $p(y|\boldsymbol{x} = \boldsymbol{x}_i^{\mathrm{tr}})$. $p(y|\boldsymbol{x})$ may be regarded as the sum of the true output $f(\boldsymbol{x})$ and noise ϵ:

$$y = f(\boldsymbol{x}) + \epsilon.$$

We assume that the noise ϵ has mean 0 and variance σ^2. This formulation is summarized in Figure 1.

Let $(\boldsymbol{x}^{\mathrm{te}}, y^{\mathrm{te}})$ be a test sample, which is not given to the user in the training phase, but will be provided in the test phase in the future. $\boldsymbol{x}^{\mathrm{te}} \in \mathcal{X}$ is a test input point following a probability distribution with density $p_{\mathrm{te}}(\boldsymbol{x})$, which is different from $p_{\mathrm{tr}}(\boldsymbol{x})$. $y^{\mathrm{te}} \in \mathcal{Y}$ is a test output value following $p(y|\boldsymbol{x} = \boldsymbol{x}^{\mathrm{te}})$, which is the same conditional density as the training phase. The goal of supervised learning is to obtain an approximation $\widehat{f}(\boldsymbol{x})$ to the true function $f(\boldsymbol{x})$ for predicting the test output value y^{te}. More formally, we would like to obtain the approximation $\widehat{f}(\boldsymbol{x})$ that minimizes the test error expected over all test samples (which is called the *generalization error*):

$$G := \mathop{\mathbb{E}}_{\boldsymbol{x}^{\mathrm{te}}} \mathop{\mathbb{E}}_{y^{\mathrm{te}}} \left[\mathrm{loss}(\widehat{f}(\boldsymbol{x}^{\mathrm{te}}), y^{\mathrm{te}}) \right],$$

where $\mathbb{E}_{\boldsymbol{x}^{\mathrm{te}}}$ denotes the expectation over $\boldsymbol{x}^{\mathrm{te}}$ drawn from $p_{\mathrm{te}}(\boldsymbol{x})$ and $\mathbb{E}_{y^{\mathrm{te}}}$ denotes the expectation over y^{te} drawn from $p(y|\boldsymbol{x} = \boldsymbol{x}^{\mathrm{te}})$. $\mathrm{loss}(\widehat{y}, y)$ is the loss function which measures the discrepancy between the true output value y and its estimate $\widehat{y}$. When the output domain $\mathcal{Y}$ is continuous, the problem is called *regression* and the *squared-loss* is often used.

$$\mathrm{loss}(\widehat{y}, y) = (\widehat{y} - y)^2.$$

On the other hand, when $\mathcal{Y} = \{+1, -1\}$, the problem is called (binary) *classification* and the *0/1-loss* is a typical choice.

$$\mathrm{loss}(\widehat{y}, y) = \begin{cases} 0 & \text{if } \mathrm{sgn}(\widehat{y}) = y, \\ 1 & \text{otherwise,} \end{cases}$$

where $\mathrm{sgn}(y) = +1$ if $y \geq 0$ and $\mathrm{sgn}(y) = -1$ if $y < 0$. Note that the 0/1-loss corresponds to the misclassification rate.

We use a parametric function $\widehat{f}(\boldsymbol{x}; \boldsymbol{\theta})$ for learning, where $\boldsymbol{\theta}$ is a parameter. A model $\widehat{f}(\boldsymbol{x}; \boldsymbol{\theta})$ is said to be *correctly specified* if there exists a parameter $\boldsymbol{\theta}^*$ such that $\widehat{f}(\boldsymbol{x}; \boldsymbol{\theta}^*) = f(\boldsymbol{x})$; otherwise the model is said to be *misspecified.* In practice, the model used for learning would be misspecified to a greater or less extent since we do not generally have enough prior knowledge for correctly specifying the model. Thus it is important to consider misspecified models when developing machine learning algorithms.

In standard supervised learning theories (Wahba, 1990; Bishop, 1995; Vapnik, 1998; Duda et al., 2001; Hastie et al., 2001; Schölkopf & Smola, 2002), the test input point $\boldsymbol{x}^{\mathrm{te}}$ is assumed to follow the same distribution as the training input point $\boldsymbol{x}^{\mathrm{tr}}$. On the other hand, in this chapter, we consider the situation called *covariate shift* (Shimodaira, 2000), i.e., the training input point $\boldsymbol{x}^{\mathrm{tr}}$ and the test input point $\boldsymbol{x}^{\mathrm{te}}$ have different distributions. Under covariate shift, most of the standard learning techniques do not work well due to the differing distributions. Below, we review recently developed techniques for mitigating the influence of covariate shift.

3 Function Learning under Covariate Shift

A standard method to learn the parameter $\boldsymbol{\theta}$ in the model $\widehat{f}(\boldsymbol{x}; \boldsymbol{\theta})$ would be *empirical risk minimization* (ERM) (Vapnik, 1998; Schölkopf & Smola, 2002):

$$\widehat{\boldsymbol{\theta}}_{\mathrm{ERM}} := \underset{\boldsymbol{\theta}}{\mathrm{argmin}} \left[\frac{1}{n_{\mathrm{tr}}} \sum_{i=1}^{n_{\mathrm{tr}}} \mathrm{loss}(\widehat{f}(\boldsymbol{x}_i^{\mathrm{tr}}; \boldsymbol{\theta}), y_i^{\mathrm{tr}}) \right].$$

If $p_{\mathrm{tr}}(\boldsymbol{x}) = p_{\mathrm{te}}(\boldsymbol{x})$, $\widehat{\boldsymbol{\theta}}_{\mathrm{ERM}}$ converges to the optimal parameter $\boldsymbol{\theta}^*$ (Shimodaira, 2000):

$$\boldsymbol{\theta}^* := \underset{\boldsymbol{\theta}}{\mathrm{argmin}}[G].$$

However, under covariate shift where $p_{\mathrm{tr}}(\boldsymbol{x}) \neq p_{\mathrm{te}}(\boldsymbol{x})$, $\widehat{\boldsymbol{\theta}}_{\mathrm{ERM}}$ does not converge to $\boldsymbol{\theta}^*$ if the model is misspecified[1].

In this section, we review various learning methods for covariate shift adaptation and show their numerical examples.

[1] $\widehat{\boldsymbol{\theta}}_{\mathrm{ERM}}$ still converges to $\boldsymbol{\theta}^*$ under covariate shift if the model is correctly specified.

3.1 Importance Weighting Techniques for Covariate Shift Adaptation

Here, we introduce various regression and classification techniques for covariate shift adaptation.

3.1.1 Importance Weighted ERM

The inconsistency of ERM is due to the difference between training and test input distributions. *Importance sampling* (Fishman, 1996) is a standard technique to compensate for the difference of distributions. The following identity shows the essence of importance sampling:

$$\mathop{\mathbb{E}}_{\boldsymbol{x}^{\mathrm{te}}}[g(\boldsymbol{x}^{\mathrm{te}})] = \int g(\boldsymbol{x}) p_{\mathrm{te}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} = \int g(\boldsymbol{x}) \frac{p_{\mathrm{te}}(\boldsymbol{x})}{p_{\mathrm{tr}}(\boldsymbol{x})} p_{\mathrm{tr}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} = \mathop{\mathbb{E}}_{\boldsymbol{x}^{\mathrm{tr}}}\left[g(\boldsymbol{x}^{\mathrm{tr}}) \frac{p_{\mathrm{te}}(\boldsymbol{x}^{\mathrm{tr}})}{p_{\mathrm{tr}}(\boldsymbol{x}^{\mathrm{tr}})}\right],$$

where $\mathbb{E}_{\boldsymbol{x}^{\mathrm{tr}}}$ and $\mathbb{E}_{\boldsymbol{x}^{\mathrm{te}}}$ denote the expectations over $\boldsymbol{x}^{\mathrm{tr}}$ and $\boldsymbol{x}^{\mathrm{te}}$ drawn from $p_{\mathrm{tr}}(\boldsymbol{x})$ and $p_{\mathrm{te}}(\boldsymbol{x})$, respectively. The quantity

$$\frac{p_{\mathrm{te}}(\boldsymbol{x})}{p_{\mathrm{tr}}(\boldsymbol{x})}$$

is called the *importance.* The above identity shows that the expectation of a function $g(\boldsymbol{x})$ over $p_{\mathrm{te}}(\boldsymbol{x})$ can be computed by the importance-weighted expectation of $g(\boldsymbol{x})$ over $p_{\mathrm{tr}}(\boldsymbol{x})$. Thus the difference of distributions can be systematically adjusted by importance weighting.

Applying the above importance weighting technique to ERM, we obtain *importance-weighted ERM* (IWERM):

$$\widehat{\boldsymbol{\theta}}_{\mathrm{IWERM}} := \mathop{\mathrm{argmin}}_{\boldsymbol{\theta}} \left[\frac{1}{n_{\mathrm{tr}}} \sum_{i=1}^{n_{\mathrm{tr}}} \frac{p_{\mathrm{te}}(\boldsymbol{x}_i^{\mathrm{tr}})}{p_{\mathrm{tr}}(\boldsymbol{x}_i^{\mathrm{tr}})} \mathrm{loss}(\widehat{f}(\boldsymbol{x}_i^{\mathrm{tr}}; \boldsymbol{\theta}), y_i^{\mathrm{tr}})\right].$$

$\widehat{\boldsymbol{\theta}}_{\mathrm{IWERM}}$ converges to $\boldsymbol{\theta}^*$ under covariate shift, even if the model is misspecified (Shimodaira, 2000). In practice, IWERM may be *regularized*, e.g., by slightly flattening the importance weight and/or adding a penalty term as

$$\mathop{\mathrm{argmin}}_{\boldsymbol{\theta}} \left[\frac{1}{n_{\mathrm{tr}}} \sum_{i=1}^{n_{\mathrm{tr}}} \left(\frac{p_{\mathrm{te}}(\boldsymbol{x}_i^{\mathrm{tr}})}{p_{\mathrm{tr}}(\boldsymbol{x}_i^{\mathrm{tr}})}\right)^{\gamma} \mathrm{loss}(\widehat{f}(\boldsymbol{x}_i^{\mathrm{tr}}; \boldsymbol{\theta}), y_i^{\mathrm{tr}}) + \lambda \boldsymbol{\theta}^{\top} \boldsymbol{\theta}\right],$$

where $0 \leq \gamma \leq 1$ is the flattening parameter, $\lambda \geq 0$ is the regularization parameter, and $^{\top}$ denotes the transpose of a matrix or a vector.

3.1.2 Importance-Weighted Regression Methods

Least-squares (LS) would be one of the most fundamental regression techniques. The importance-weighted regression method for the squared-loss (see Figure 2), called

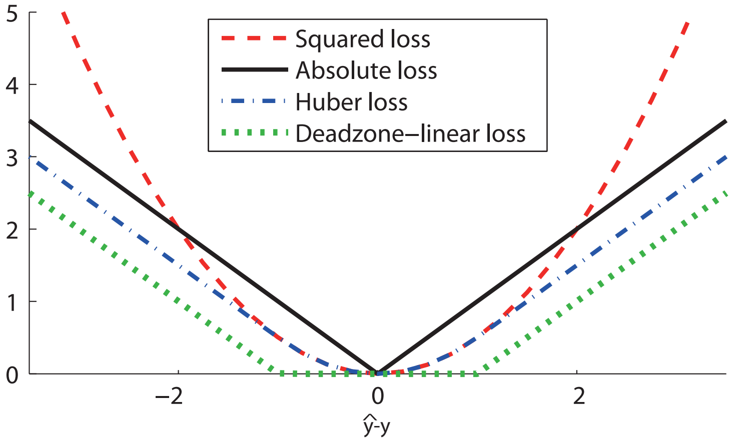


Figure 2: Loss functions for regression. y is the true output value at an input point and $\widehat{y}$ is its estimate.

importance-weighted LS (IWLS), is given as follows:

$$\widehat{\boldsymbol{\theta}}_{\mathrm{IWLS}} := \underset{\boldsymbol{\theta}}{\mathrm{argmin}} \left[\frac{1}{n_{\mathrm{tr}}} \sum_{i=1}^{n_{\mathrm{tr}}} \left(\frac{p_{\mathrm{te}}(\boldsymbol{x}_i^{\mathrm{tr}})}{p_{\mathrm{tr}}(\boldsymbol{x}_i^{\mathrm{tr}})} \right)^{\gamma} \left(\widehat{f}(\boldsymbol{x}_i^{\mathrm{tr}}; \boldsymbol{\theta}) - y_i^{\mathrm{tr}} \right)^2 + \lambda \boldsymbol{\theta}^{\top} \boldsymbol{\theta} \right]. \tag{1}$$

Let us employ the following linear model:

$$\widehat{f}(\boldsymbol{x}; \boldsymbol{\theta}) = \sum_{\ell=1}^{b} \theta_{\ell} \phi_{\ell}(\boldsymbol{x}), \tag{2}$$

where $\{\phi_{\ell}(\boldsymbol{x})\}_{\ell=1}^{b}$ are fixed linearly-independent basis functions. Then the solution $\widehat{\boldsymbol{\theta}}_{\mathrm{IWLS}}$ is given *analytically* as

$$\widehat{\boldsymbol{\theta}}_{\mathrm{IWLS}} = (\boldsymbol{X}^{\mathrm{tr}\top} \boldsymbol{W}^{\gamma} \boldsymbol{X}^{\mathrm{tr}} + n_{\mathrm{tr}} \lambda \boldsymbol{I}_b)^{-1} \boldsymbol{X}^{\mathrm{tr}\top} \boldsymbol{W}^{\gamma} \boldsymbol{y}^{\mathrm{tr}}, \tag{3}$$

where $\boldsymbol{X}^{\mathrm{tr}}$ is the *design matrix*, i.e., $\boldsymbol{X}^{\mathrm{tr}}$ is the $n_{\mathrm{tr}} \times b$ matrix with the (i, ℓ)-th element $X_{i,\ell}^{\mathrm{tr}} = \phi_{\ell}(\boldsymbol{x}_i^{\mathrm{tr}})$, $\boldsymbol{W}$ is the diagonal matrix with the i-th diagonal element $\frac{p_{\mathrm{te}}(\boldsymbol{x}_i^{\mathrm{tr}})}{p_{\mathrm{tr}}(\boldsymbol{x}_i^{\mathrm{tr}})}$, $\boldsymbol{I}_b$ is the b-dimensional identity matrix, and $\boldsymbol{y}^{\mathrm{tr}}$ is the n_{tr}-dimensional vector with the i-th element y_i^{tr}.

The LS method often suffers from excessive sensitivity to *outliers* (i.e., irregular values) and less reliability. A popular alternative is *importance-weighted least absolute* (IWLA) regression—instead of the squared loss, the absolute loss is used (see Figure 2):

$$\widehat{\boldsymbol{\theta}}_{\mathrm{IWLA}} = \underset{\boldsymbol{\theta}}{\mathrm{argmin}} \left[\frac{1}{n_{\mathrm{tr}}} \sum_{i=1}^{n_{\mathrm{tr}}} \left(\frac{p_{\mathrm{te}}(\boldsymbol{x}_i^{\mathrm{tr}})}{p_{\mathrm{tr}}(\boldsymbol{x}_i^{\mathrm{tr}})} \right)^{\gamma} \left| \widehat{f}(\boldsymbol{x}_i^{\mathrm{tr}}; \boldsymbol{\theta}) - y_i^{\mathrm{tr}} \right| + \lambda \boldsymbol{\theta}^{\top} \boldsymbol{\theta} \right].$$

For the linear model (2), the above optimization problem is reduced to a quadratic program, which can be solved by a standard optimization software. If the regularization term

$\boldsymbol{\theta}^\top\boldsymbol{\theta}$ is replaced by the ℓ_1-regularizer $\sum_{\ell=1}^{b}|\theta_\ell|$, the optimization problem is reduced to a linear program, which may be solved more efficiently. Furthermore, the ℓ_1-regularizer is known to induce a *sparse* solution (Williams, 1995; Tibshirani, 1996; Chen et al., 1998).

Although the LA method is robust against outliers, it tends to have a large variance when the noise is Gaussian. The use of the *Huber loss* can mitigate this problem:

$$\widehat{\boldsymbol{\theta}}_{\text{Huber}} = \underset{\boldsymbol{\theta}}{\text{argmin}}\left[\frac{1}{n_{\text{tr}}}\sum_{i=1}^{n_{\text{tr}}}\left(\frac{p_{\text{te}}(\boldsymbol{x}_i^{\text{tr}})}{p_{\text{tr}}(\boldsymbol{x}_i^{\text{tr}})}\right)^{\gamma}\rho_\tau\left(\widehat{f}(\boldsymbol{x}_i^{\text{tr}};\boldsymbol{\theta}) - y_i^{\text{tr}}\right) + \lambda\boldsymbol{\theta}^\top\boldsymbol{\theta}\right],$$

where τ (≥ 0) is the robustness parameter and ρ_τ is the Huber loss defined as follows (see Figure 2):

$$\rho_\tau(y) := \begin{cases} \frac{1}{2}y^2 & \text{if } |y| \leq \tau, \\ \tau|y| - \frac{1}{2}\tau^2 & \text{if } |y| > \tau. \end{cases}$$

Thus, the squared loss is applied to 'good' samples with small fitting error, and the absolute loss is applied to 'bad' samples with large fitting error. The above optimization problem can be reduced to a quadratic program (Mangasarian & Musicant, 2000), which can be solved by a standard optimization software.

Another variant of the IWLA is *importance-weighted support vector regression* (IWSVR):

$$\widehat{\boldsymbol{\theta}}_{\text{SVR}} = \underset{\boldsymbol{\theta}}{\text{argmin}}\left[\frac{1}{n_{\text{tr}}}\sum_{i=1}^{n_{\text{tr}}}\left(\frac{p_{\text{te}}(\boldsymbol{x}_i^{\text{tr}})}{p_{\text{tr}}(\boldsymbol{x}_i^{\text{tr}})}\right)^{\gamma}\left|\widehat{f}(\boldsymbol{x}_i^{\text{tr}};\boldsymbol{\theta}) - y_i^{\text{tr}}\right|_{\epsilon} + \lambda\boldsymbol{\theta}^\top\boldsymbol{\theta}\right],$$

where $|\cdot|_\epsilon$ is the *deadzone-linear loss* (or *Vapnik's ϵ-insensitive loss*) defined as follows (see Figure 2):

$$|x|_\epsilon := \begin{cases} 0 & \text{if } |x| \leq \epsilon, \\ |x| - \epsilon & \text{if } |x| > \epsilon. \end{cases}$$

For the linear model (2), the above optimization problem is reduced to a quadratic program (Vapnik, 1998), which can be solved by a standard optimization software. If the regularization term $\boldsymbol{\theta}^\top\boldsymbol{\theta}$ is replaced by the ℓ_1-regularizer $\sum_{\ell=1}^{b}|\theta_\ell|$, the optimization problem is reduced to a linear program.

3.1.3 Importance-Weighted Classification Methods

In the binary classification scenario where $\mathcal{Y} = \{+1, -1\}$, *Fisher discriminant analysis* (FDA) (Fisher, 1936), *logistic regression* (LR) (Hastie et al., 2001), *support vector machine* (SVM) (Vapnik, 1998; Schölkopf & Smola, 2002), and *boosting* (Freund & Schapire, 1996; Breiman, 1998; Friedman et al., 2000) would be popular learning algorithms. They can be regarded as ERM methods with different loss functions (see Figure 3).

An importance-weighted version of FDA, IWFDA, is given by

$$\widehat{\boldsymbol{\theta}}_{\text{IWFDA}} := \underset{\boldsymbol{\theta}}{\text{argmin}}\left[\frac{1}{n_{\text{tr}}}\sum_{i=1}^{n_{\text{tr}}}\left(\frac{p_{\text{te}}(\boldsymbol{x}_i^{\text{tr}})}{p_{\text{tr}}(\boldsymbol{x}_i^{\text{tr}})}\right)^{\gamma}\left(1 - y_i^{\text{tr}}\widehat{f}(\boldsymbol{x}_i^{\text{tr}};\boldsymbol{\theta})\right)^2 + \lambda\boldsymbol{\theta}^\top\boldsymbol{\theta}\right],$$

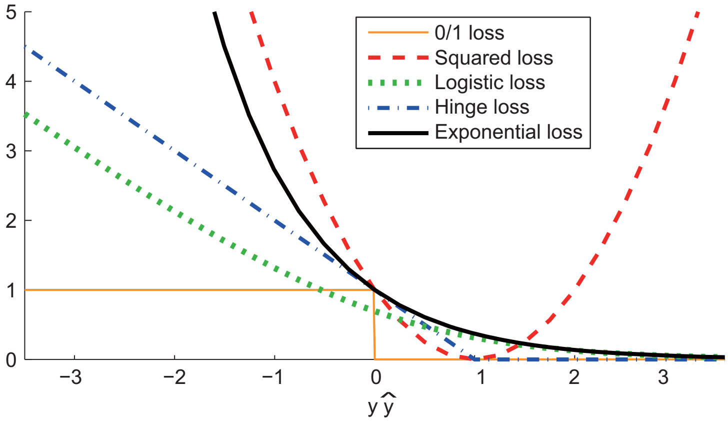


Figure 3: Loss functions for classification. y is the true output value at an input point and $\widehat{y}$ is its estimate.

which is essentially equivalent to Eq.(1) since $(y_i^{\mathrm{tr}})^2 = 1$.

An importance-weighted version of LR, IWLR, is given by

$$\widehat{\boldsymbol{\theta}}_{\mathrm{IWLR}} := \mathop{\mathrm{argmin}}_{\boldsymbol{\theta}} \left[\sum_{i=1}^{n_{\mathrm{tr}}} \left(\frac{p_{\mathrm{te}}(\boldsymbol{x}_i^{\mathrm{tr}})}{p_{\mathrm{tr}}(\boldsymbol{x}_i^{\mathrm{tr}})} \right)^{\gamma} \log \left(1 + \exp \left(-y_i^{\mathrm{tr}} \widehat{f}(\boldsymbol{x}_i^{\mathrm{tr}}; \boldsymbol{\theta}) \right) \right) + \lambda \boldsymbol{\theta}^\top \boldsymbol{\theta} \right],$$

which is usually solved by (quasi-)Newton methods.

An importance-weighted version of SVM, IWSVM, is given by

$$\widehat{\boldsymbol{\theta}}_{\mathrm{IWSVM}} := \mathop{\mathrm{argmin}}_{\boldsymbol{\theta}} \left[\sum_{i=1}^{n_{\mathrm{tr}}} \left(\frac{p_{\mathrm{te}}(\boldsymbol{x}_i^{\mathrm{tr}})}{p_{\mathrm{tr}}(\boldsymbol{x}_i^{\mathrm{tr}})} \right)^{\gamma} \max \left(0, 1 - y_i^{\mathrm{tr}} \widehat{f}(\boldsymbol{x}_i^{\mathrm{tr}}; \boldsymbol{\theta}) \right) + \lambda \boldsymbol{\theta}^\top \boldsymbol{\theta} \right],$$

whose solution can be obtained by a standard quadratic programming solver.

An importance-weighted version of Boosting, IWB, is given by

$$\widehat{\boldsymbol{\theta}}_{\mathrm{IWB}} := \mathop{\mathrm{argmin}}_{\boldsymbol{\theta}} \left[\sum_{i=1}^{n_{\mathrm{tr}}} \left(\frac{p_{\mathrm{te}}(\boldsymbol{x}_i^{\mathrm{tr}})}{p_{\mathrm{tr}}(\boldsymbol{x}_i^{\mathrm{tr}})} \right)^{\gamma} \exp \left(-y_i^{\mathrm{tr}} \widehat{f}(\boldsymbol{x}_i^{\mathrm{tr}}; \boldsymbol{\theta}) \right) + \lambda \boldsymbol{\theta}^\top \boldsymbol{\theta} \right],$$

which is usually solved by stage-wise optimization.

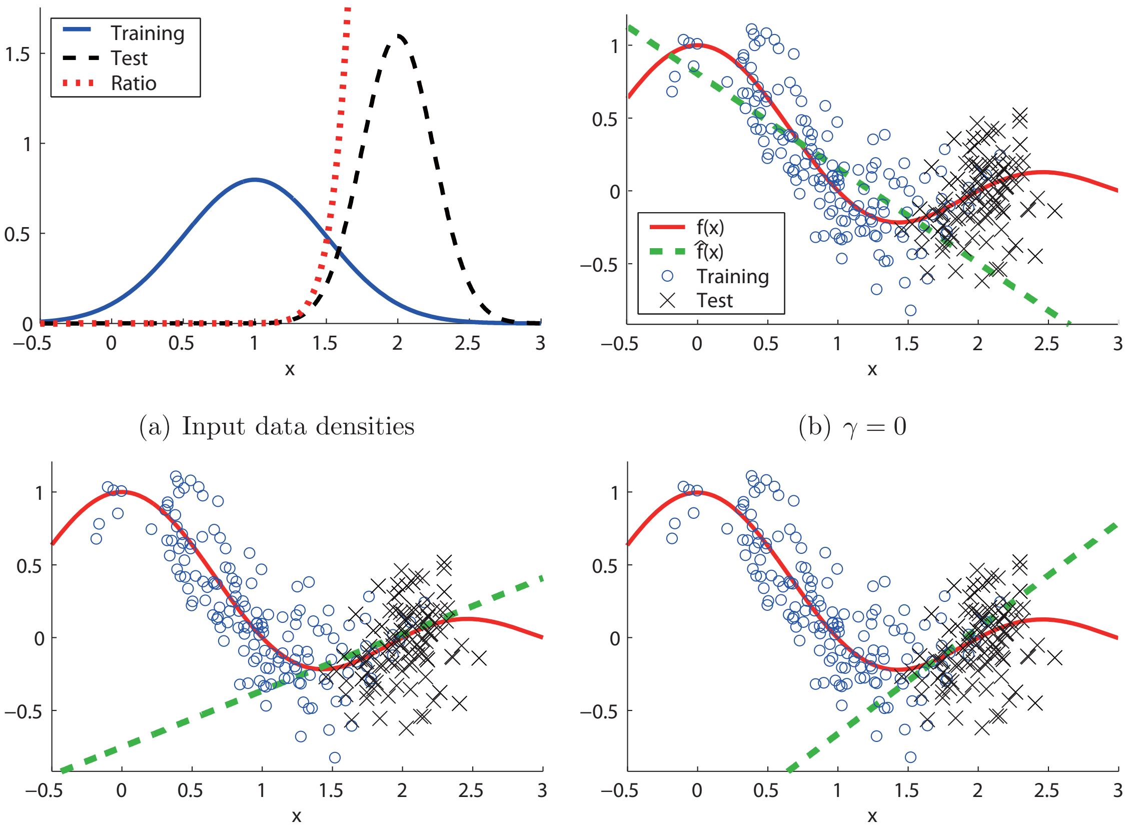
3.2 Numerical Examples

Here we illustrate the behavior of IWERM using toy regression and classification data sets.

3.2.1 Regression

Let us consider one-dimensional regression problem. Let the learning target function be $f(x) = \mathrm{sinc}(x)$, and let the training and test input densities be

$$p_{\mathrm{tr}}(x) = N(x; 1, (1/2)^2) \ \text{ and } \ p_{\mathrm{te}}(x) = N(x; 2, (1/4)^2),$$



(a) Input data densities

(b) $\gamma = 0$

(c) $\gamma = 0.5$

(d) $\gamma = 1$

Figure 4: An illustrative regression example with covariate shift. (a) The probability density functions of the training and test input points and their ratio (i.e., the importance). (b)–(d) The learning target function $f(x)$ (the solid line), training samples ('$\circ$'), a learned function $\widehat{f}(x)$ (the dashed line), and test samples ('$\times$').

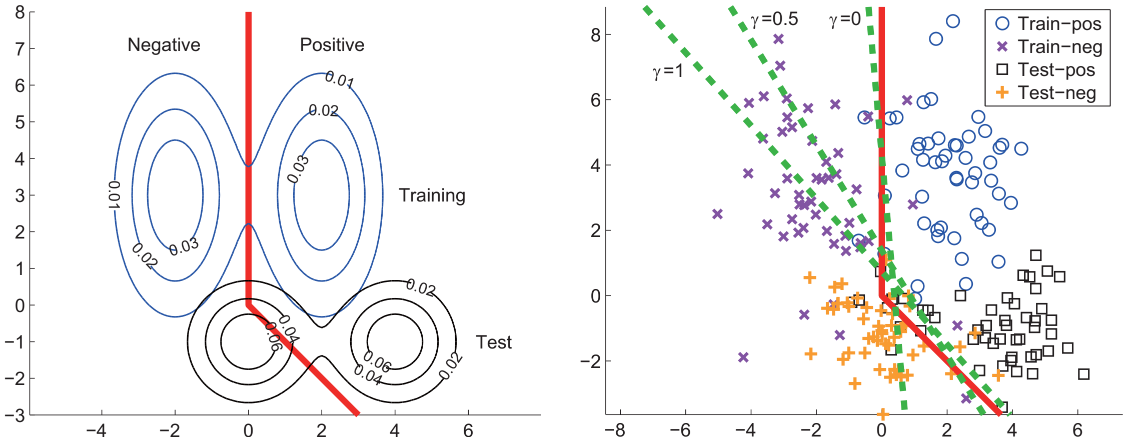
where $N(x;\mu,\sigma^2)$ denotes the Gaussian density with mean μ and variance σ^2. As illustrated in Figure 4(a), we are considering a (weak) extrapolation problem since the training input points are distributed in the left-hand side of the input domain and the test input points are distributed in the right-hand side.

We create the training output value $\{y_i^{\mathrm{tr}}\}_{i=1}^{n_{\mathrm{tr}}}$ as $y_i^{\mathrm{tr}} = f(x_i^{\mathrm{tr}}) + \epsilon_i^{\mathrm{tr}}$, where $\{\epsilon_i^{\mathrm{tr}}\}_{i=1}^{n_{\mathrm{tr}}}$ are i.i.d. noise drawn from $N(\epsilon;0,(1/4)^2)$. Let the number of training samples be $n_{\mathrm{tr}} = 150$, and we use the following linear model:

$$\widehat{f}(x;\boldsymbol{\theta}) = \theta_1 x + \theta_2.$$

The parameter $\boldsymbol{\theta}$ is learned by IWLS.

Here we fix the regularization parameter to $\lambda = 0$, and compare the performance of IWLS for $\gamma = 0, 0.5, 1$. When $\gamma = 0$, a good approximation of the left-hand side of the sinc function can be obtained (see Figure 4(b)). However, this is not appropriate



(a) Optimal decision boundary (the thick solid line) and contours of training and test input densities (thin solid lines).

(b) Optimal decision boundary (solid line) and learned boundaries (dashed lines). '○' and '×' denote the positive and negative training samples, while '□' and '+' denote the positive and negative test samples.

Figure 5: An illustrative classification example with covariate shift.

for estimating the test output values ('×' in the figure). Thus, IWLS with $\gamma = 0$ (i.e., ordinary LS) results in a large test error. Figure 4(d) depicts the learned function when $\gamma = 1$, which tends to approximate the test output values well, but having a large variance. Figure 4(c) depicts a learned function when $\gamma = 0.5$, which yields even better estimation of the test output values for this particular data realization.

3.2.2 Classification

Let us consider a binary classification problem on the two-dimensional input space. Let the class posterior probabilities given input $\boldsymbol{x}$ be

$$p(y = +1 \mid \boldsymbol{x}) = \frac{1}{2}\left(1 + \tanh\left(x^{(1)} + \min(0, x^{(2)})\right)\right), \tag{4}$$

where $\boldsymbol{x} = (x^{(1)}, x^{(2)})^\top$ and $p(y = -1 \mid \boldsymbol{x}) = 1 - p(y = +1 \mid \boldsymbol{x})$. The optimal decision boundary, i.e., a set of all $\boldsymbol{x}$ such that $p(y = +1 \mid \boldsymbol{x}) = p(y = -1 \mid \boldsymbol{x}) = 1/2$ is illustrated in Figure 5(a).

Let the training and test input densities be

$$p_{\mathrm{tr}}(\boldsymbol{x}) = \frac{1}{2}N\left(\boldsymbol{x}; \begin{bmatrix} -2 \\ 3 \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 4 \end{bmatrix}\right) + \frac{1}{2}N\left(\boldsymbol{x}; \begin{bmatrix} 2 \\ 3 \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 4 \end{bmatrix}\right),$$
$$p_{\mathrm{te}}(\boldsymbol{x}) = \frac{1}{2}N\left(\boldsymbol{x}; \begin{bmatrix} 0 \\ -1 \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\right) + \frac{1}{2}N\left(\boldsymbol{x}; \begin{bmatrix} 4 \\ -1 \end{bmatrix}, \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}\right),$$

where $N(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ is the multivariate Gaussian density with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$. This setup implies that we are considering a (weak) extrapolation problem. Contours of the training and test input densities are illustrated in Figure 5(a).

Let the number of training samples be $n_{\mathrm{tr}} = 500$, and we create training input points $\{\boldsymbol{x}_i^{\mathrm{tr}}\}_{i=1}^{n_{\mathrm{tr}}}$ following $p_{\mathrm{tr}}(\boldsymbol{x})$ and training output labels $\{y_i^{\mathrm{tr}}\}_{i=1}^{n_{\mathrm{tr}}}$ following $p(y|\boldsymbol{x} = \boldsymbol{x}_i^{\mathrm{tr}})$. Similarly, let the number of test samples be $n_{\mathrm{te}} = 500$, and we create n_{te} test input points $\{\boldsymbol{x}_j^{\mathrm{te}}\}_{j=1}^{n_{\mathrm{te}}}$ following $p_{\mathrm{te}}(\boldsymbol{x})$ and test output labels $\{y_j^{\mathrm{te}}\}_{j=1}^{n_{\mathrm{te}}}$ following $p(y|\boldsymbol{x} = \boldsymbol{x}_j^{\mathrm{te}})$. We use the following linear model:

$$\widehat{f}(\boldsymbol{x}; \boldsymbol{\theta}) = \theta_1 x^{(1)} + \theta_2 x^{(2)} + \theta_3.$$

The parameter $\boldsymbol{\theta}$ is learned by IWFDA.

Here we fix the regularization parameter to $\lambda = 0$, and compare the performance of IWFDA for $\gamma = 0, 0.5, 1$. Figure 5(b) depicts an example of realizations of training and test samples, and decision boundaries obtained by IWFDA. For this particular realization of data samples, $\gamma = 0.5$ or 1 works better than $\gamma = 0$.

4 Model Selection under Covariate Shift

As illustrated in the previous section, importance-weighting is a promising approach to covariate shift adaptation, given that the flattening parameter γ is chosen appropriately. Although $\gamma = 0.5$ worked well both for the toy regression and classification experiments in the previous section, $\gamma = 0.5$ may not always be the best choice. Indeed, an appropriate value of γ depends on the learning target function, models, the noise level in the training samples, etc. Therefore, *model selection* needs to be appropriately carried out for enhancing the generalization capability under covariate shift.

The goal of model selection is to determine the model (e.g, basis functions, the flattening parameter γ, and the regularization parameter λ) so that the generalization error is minimized (Akaike, 1970; Mallows, 1973; Akaike, 1974; Takeuchi, 1976; Schwarz, 1978; Rissanen, 1978; Craven & Wahba, 1979; Akaike, 1980; Rissanen, 1987; Shibata, 1989; Wahba, 1990; Efron & Tibshirani, 1993; Murata et al., 1994; Konishi & Kitagawa, 1996; Ishiguro et al., 1997; Vapnik, 1998; Sugiyama & Ogawa, 2001; Sugiyama & Müller, 2002; Sugiyama et al., 2004). The true generalization error is not accessible since it contains the unknown learning target function. Thus, some generalization error estimator needs to be used instead. However, standard generalization error estimators such as *cross-validation* (CV) are heavily biased under covariate shift, and therefore they are no longer reliable. In this section, we review a modified CV method that possesses proper unbiasedness even under covariate shift.

4.1 Importance-Weighted Cross-Validation

One of the popular techniques for estimating the generalization error is CV (Stone, 1974; Wahba, 1990). CV has been shown to give an *almost* unbiased estimate of the generalization error with finite samples (Luntz & Brailovsky, 1969; Schölkopf & Smola, 2002). However, such almost unbiasedness is no longer fulfilled under covariate shift.

To cope with this problem, a variant of CV called *importance-weighted CV* (IWCV) has been proposed (Sugiyama et al., 2007). Let us randomly divide the training set $\mathcal{Z} = \{(\boldsymbol{x}_i^{\mathrm{tr}}, y_i^{\mathrm{tr}})\}_{i=1}^{n_{\mathrm{tr}}}$ into k disjoint non-empty subsets $\{\mathcal{Z}_i\}_{i=1}^{k}$ of (approximately) the same size. Let $\widehat{f}_{\mathcal{Z}_i}(\boldsymbol{x})$ be a function learned from $\{\mathcal{Z}_{i'}\}_{i' \neq i}$ (i.e., without $\mathcal{Z}_i$). Then the *k-fold IWCV* (kIWCV) estimate of the generalization error G is given by

$$\widehat{G}_{k\mathrm{IWCV}} = \frac{1}{k}\sum_{i=1}^{k}\frac{1}{|\mathcal{Z}_i|}\sum_{(\boldsymbol{x},y)\in\mathcal{Z}_i}\frac{p_{\mathrm{te}}(\boldsymbol{x})}{p_{\mathrm{tr}}(\boldsymbol{x})}\mathrm{loss}(\widehat{f}_{\mathcal{Z}_i}(\boldsymbol{x}), y),$$

where $|\mathcal{Z}_i|$ is the number of samples in the subset $\mathcal{Z}_i$.

When $k = n_{\mathrm{tr}}$, kIWCV is particularly called *IW leave-one-out CV* (IWLOOCV):

$$\widehat{G}_{\mathrm{IWLOOCV}} = \frac{1}{n_{\mathrm{tr}}}\sum_{i=1}^{n_{\mathrm{tr}}}\frac{p_{\mathrm{te}}(\boldsymbol{x}_i^{\mathrm{tr}})}{p_{\mathrm{tr}}(\boldsymbol{x}_i^{\mathrm{tr}})}\mathrm{loss}(\widehat{f}_i(\boldsymbol{x}_i^{\mathrm{tr}}), y_i^{\mathrm{tr}}),$$

where $\widehat{f}_i(\boldsymbol{x})$ is a function learned from $\{(\boldsymbol{x}_{i'}^{\mathrm{tr}}, y_{i'}^{\mathrm{tr}})\}_{i' \neq i}$ (i.e., without $(\boldsymbol{x}_i^{\mathrm{tr}}, y_i^{\mathrm{tr}})$). It was proved that IWLOOCV gives an *almost* unbiased estimate of the generalization error even under covariate shift (Sugiyama et al., 2007). More precisely, IWLOOCV for n_{tr} training samples gives an unbiased estimate of the generalization error for $n_{\mathrm{tr}} - 1$ training samples:

$$\mathop{\mathbb{E}}_{\{\boldsymbol{x}_i^{\mathrm{tr}}\}_{i=1}^{n_{\mathrm{tr}}}}\mathop{\mathbb{E}}_{\{y_i^{\mathrm{tr}}\}_{i=1}^{n_{\mathrm{tr}}}}\left[\widehat{G}_{\mathrm{IWLOOCV}}\right] = \mathop{\mathbb{E}}_{\{\boldsymbol{x}_i^{\mathrm{tr}}\}_{i=1}^{n_{\mathrm{tr}}}}\mathop{\mathbb{E}}_{\{y_i^{\mathrm{tr}}\}_{i=1}^{n_{\mathrm{tr}}}}[G'] \approx \mathop{\mathbb{E}}_{\{\boldsymbol{x}_i^{\mathrm{tr}}\}_{i=1}^{n_{\mathrm{tr}}}}\mathop{\mathbb{E}}_{\{y_i^{\mathrm{tr}}\}_{i=1}^{n_{\mathrm{tr}}}}[G],$$

where $\mathbb{E}_{\{\boldsymbol{x}_i^{\mathrm{tr}}\}_{i=1}^{n_{\mathrm{tr}}}}$ denotes the expectation over $\{\boldsymbol{x}_i^{\mathrm{tr}}\}_{i=1}^{n_{\mathrm{tr}}}$ drawn i.i.d. from $p_{\mathrm{tr}}(\boldsymbol{x})$, $\mathbb{E}_{\{y_i^{\mathrm{tr}}\}_{i=1}^{n_{\mathrm{tr}}}}$ denotes the expectation over $\{y_i^{\mathrm{tr}}\}_{i=1}^{n_{\mathrm{tr}}}$ each drawn from $p(y|\boldsymbol{x} = \boldsymbol{x}_i^{\mathrm{tr}})$, and G' denotes the generalization error for $n_{\mathrm{tr}} - 1$ training samples. A similar proof is also possible for kIWCV, but the bias is slightly larger (Hastie et al., 2001).

The almost unbiasedness of IWCV holds for any loss function, any model, and any parameter learning method; even non-identifiable models (Watanabe, 2009) or non-parametric learning methods (Schölkopf & Smola, 2002) are allowed. Thus IWCV is a highly flexible model selection technique under covariate shift. For other model selection criteria under covariate shift, see Shimodaira (2000) for regular models with smooth losses and Sugiyama and Müller (2005) for linear models with the squared loss.

4.2 Numerical Examples

Here we illustrate the behavior of IWCV using the same toy data sets as Section 3.2.

4.2.1 Regression

Let us continue the one-dimensional regression simulation in Section 3.2.1.

As illustrated in Figure 4 in Section 3.2.1, IWLS with flattening parameter $\gamma = 0.5$ appears to work well for that particular realization of data samples. However, the best value of γ would depend on the realization of samples. In order to investigate this systematically, let us repeat the simulation 1000 times with different random seeds, i.e., in each run $\{(x_i^{\mathrm{tr}}, \epsilon_i^{\mathrm{tr}})\}_{i=1}^{n_{\mathrm{tr}}}$ are randomly drawn and the scores of 10-fold IWCV and 10-fold ordinary CV are calculated for $\gamma = 0, 0.1, 0.2, \ldots, 1$. The means and standard deviations of the generalization error G and its estimate by each method are depicted as functions of γ in Figure 6. The graphs show that IWCV gives very accurate unbiased estimates of the generalization error, while ordinary CV is heavily biased.

Next we investigate the model selection performance. The flattening parameter γ is chosen from $\{0, 0.1, 0.2, \ldots, 1\}$ so that the score of each model selection criterion is minimized. The mean and standard deviation of the generalization error G of the learned function obtained by each method over 1000 runs are described in Table 1. This shows that IWCV gives significantly smaller generalization errors than ordinary CV, under the *t-test* (Henkel, 1976) at the significance level 5%. For reference, the generalization error when the flattening parameter γ is chosen optimally (i.e., in each trial, γ is chosen so that the true generalization error is minimized) is described as 'Optimal' in the table. The result shows that the performance of IWCV is rather close to that of the optimal choice.

4.2.2 Classification

Let us continue the toy classification simulation in Section 3.2.2.

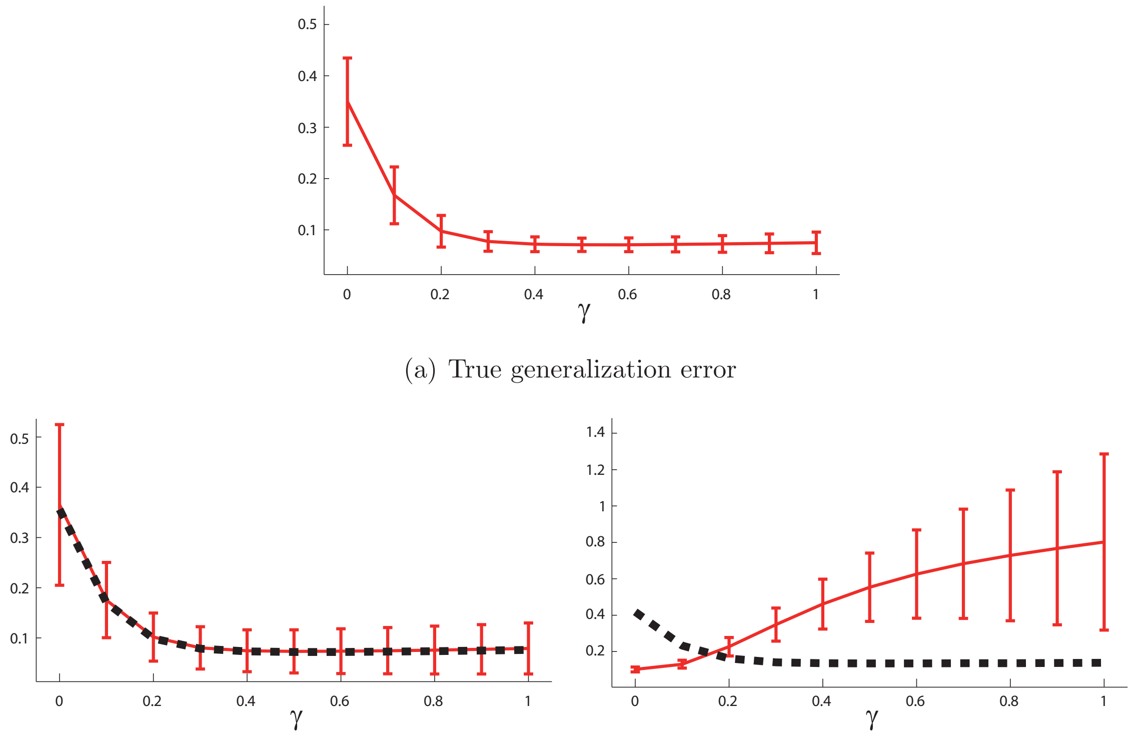
In Figure 5(b) in Section 3.2.2, IWFDA with a middle/large flattening parameter γ appears to work well for that particular realization of samples. Here, we investigate the choice of the flattening parameter value by IWCV and ordinary CV. Figure 7 depicts the means and standard deviations of the generalization error G (i.e., the misclassification rate) and its estimate by each method over 1000 runs, as functions of the flattening parameter γ in IWFDA. The graphs clearly show that IWCV gives much better estimates of the generalization error than ordinary CV.

Next we investigate the model selection performance. The flattening parameter γ is chosen from $\{0, 0.1, 0.2, \ldots, 1\}$ so that the score of each model selection criterion is minimized. The mean and standard deviation of the generalization error G of the learned function obtained by each method over 1000 runs are described in Table 2. The table shows that IWCV gives significantly smaller test errors than ordinary CV, and the performance of IWCV is rather close to that of the optimal choice.

5 Importance Estimation

In the previous sections, we have seen that the importance weight

$$w(\boldsymbol{x}) = \frac{p_{\mathrm{te}}(\boldsymbol{x})}{p_{\mathrm{tr}}(\boldsymbol{x})}$$



(a) True generalization error

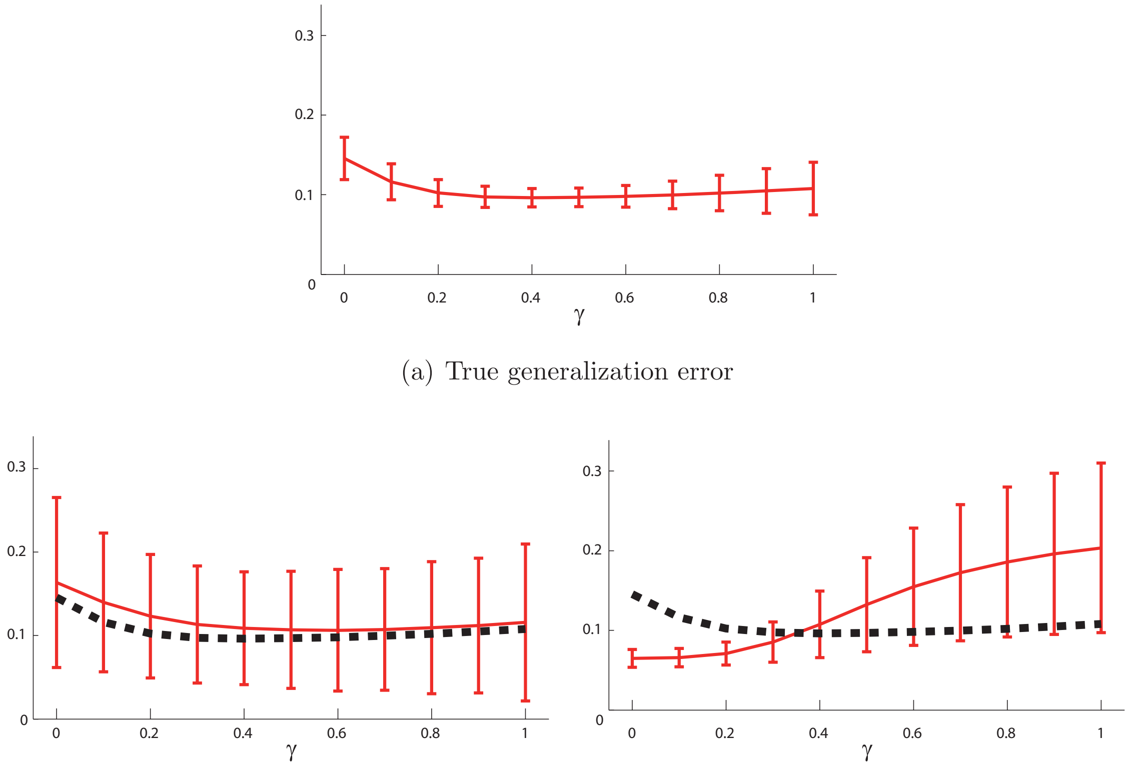
(b) IWCV score

(c) Ordinary CV score

Figure 6: Generalization error and its estimates obtained by IWCV and ordinary CV as functions of the flattening parameter γ in IWLS for the regression examples in Figure 4. Thick dashed curves in the bottom graphs depict the true generalization error for clear comparison.

Table 1: The mean and standard deviation of the generalization error G obtained by each method for the toy regression data set. The best method and comparable ones by the t-test at the significance level 5% are indicated by '$\circ$'. For reference, the generalization error obtained with the optimal γ (i.e., the minimum generalization error) is described as 'Optimal'.

IWCV	Ordinary CV	Optimal
$^{\circ}0.077 \pm 0.020$	0.356 ± 0.086	0.069 ± 0.011



(a) True generalization error

(b) IWCV score

(c) Ordinary CV score

Figure 7: The generalization error G (i.e., the misclassification rate) and its estimates obtained by IWCV and ordinary CV as functions of the flattening parameter γ in IWFDA for the toy classification examples in Figure 5. Thick dashed curves in the bottom graphs depict the true generalization error for clear comparison.

Table 2: The mean and standard deviation of the generalization error G (i.e., the misclassification rate) obtained by each method for the toy classification data set. The best method and comparable ones by the t-test at the significance level 5% are indicated by '$\circ$'. For reference, the generalization error obtained with the optimal γ (i.e., the minimum generalization error) is described as 'Optimal'.

IWCV	Ordinary CV	Optimal
$^\circ 0.108 \pm 0.027$	0.131 ± 0.029	0.091 ± 0.009

plays a central role in covariate shift adaptation. However, the importance weight is unknown in practice and needs to be estimated from data. In this section, we review importance estimation methods.

Here we assume that in addition to the training input samples $\{\boldsymbol{x}_i^{\mathrm{tr}}\}_{i=1}^{n_{\mathrm{tr}}}$ drawn independently from $p_{\mathrm{tr}}(\boldsymbol{x})$, we are given test input samples $\{\boldsymbol{x}_j^{\mathrm{te}}\}_{j=1}^{n_{\mathrm{te}}}$ drawn independently from $p_{\mathrm{te}}(\boldsymbol{x})$. Thus the goal of the importance estimation problem addressed here is to estimate the importance function $w(\boldsymbol{x})$ from $\{\boldsymbol{x}_i^{\mathrm{tr}}\}_{i=1}^{n_{\mathrm{tr}}}$ and $\{\boldsymbol{x}_j^{\mathrm{te}}\}_{j=1}^{n_{\mathrm{te}}}$.

5.1 Kernel Density Estimation

Kernel density estimation (KDE) is a non-parametric technique to estimate a probability density function $p(\boldsymbol{x})$ from its i.i.d. samples $\{\boldsymbol{x}_i\}_{i=1}^{n}$. For the Gaussian kernel

$$K_\sigma(\boldsymbol{x},\boldsymbol{x}') = \exp\left(-\frac{\|\boldsymbol{x}-\boldsymbol{x}'\|^2}{2\sigma^2}\right), \tag{5}$$

KDE is expressed as

$$\widehat{p}(\boldsymbol{x}) = \frac{1}{n_{\mathrm{tr}}(2\pi\sigma^2)^{d/2}} \sum_{\ell=1}^{n} K_\sigma(\boldsymbol{x},\boldsymbol{x}_\ell).$$

The performance of KDE depends on the choice of the kernel width σ. It can be optimized by cross-validation (CV) as follows (Härdle et al., 2004): First, divide the samples $\{\boldsymbol{x}_i\}_{i=1}^{n}$ into k disjoint non-empty subsets $\{\mathcal{X}_r\}_{r=1}^{k}$ of (approximately) the same size. Then obtain a density estimator $\widehat{p}_{\mathcal{X}_r}(\boldsymbol{x})$ from $\{\mathcal{X}_i\}_{i\neq r}$ (i.e., without $\mathcal{X}_r$), and compute its log-likelihood for the hold-out subset $\mathcal{X}_r$:

$$\frac{1}{|\mathcal{X}_r|}\sum_{\boldsymbol{x}\in\mathcal{X}_r} \log \widehat{p}_{\mathcal{X}_r}(\boldsymbol{x}),$$

where $|\mathcal{X}|$ denotes the number of elements in the set $\mathcal{X}$. Repeat this procedure for $r = 1, 2, \ldots, k$ and choose the value of σ such that the average of the above hold-out log-likelihood over all r is maximized. Note that the average hold-out log-likelihood is an almost unbiased estimate of the Kullback-Leibler divergence from $p(\boldsymbol{x})$ to $\widehat{p}(\boldsymbol{x})$, up to an irrelevant constant.

KDE can be used for importance estimation by first obtaining density estimators $\widehat{p}_{\mathrm{tr}}(\boldsymbol{x})$ and $\widehat{p}_{\mathrm{te}}(\boldsymbol{x})$ separately from $\{\boldsymbol{x}_i^{\mathrm{tr}}\}_{i=1}^{n_{\mathrm{tr}}}$ and $\{\boldsymbol{x}_j^{\mathrm{te}}\}_{j=1}^{n_{\mathrm{te}}}$, and then estimating the importance by $\widehat{w}(\boldsymbol{x}) = \widehat{p}_{\mathrm{te}}(\boldsymbol{x})/\widehat{p}_{\mathrm{tr}}(\boldsymbol{x})$. However, division by an estimated density can magnify the estimation error, so directly estimating the importance weight in a single-shot process would be more preferable.

5.2 Kullback-Leibler Importance Estimation Procedure

The *Kullback-Leibler importance estimation procedure* (KLIEP) (Sugiyama et al., 2008) directly gives an estimate of the importance function without going through density estimation by matching the two densities $p_{\mathrm{tr}}(\boldsymbol{x})$ and $p_{\mathrm{te}}(\boldsymbol{x})$ in terms of the *Kullback-Leibler divergence* (Kullback & Leibler, 1951).

Let us model the importance weight $w(\boldsymbol{x})$ by the following kernel model:

$$\widehat{w}(\boldsymbol{x}) = \sum_{\ell=1}^{n_{\mathrm{te}}} \alpha_\ell K_\sigma(\boldsymbol{x}, \boldsymbol{x}_\ell^{\mathrm{te}}),$$

where $\boldsymbol{\alpha} = (\alpha_1, \alpha_2, \ldots, \alpha_{n_{\mathrm{te}}})^\top$ are parameters to be learned from data samples and $K_\sigma(\boldsymbol{x}, \boldsymbol{x}')$ is the Gaussian kernel (see Eq.(5)). An estimate of the density $p_{\mathrm{te}}(\boldsymbol{x})$ is given by using the model $\widehat{w}(\boldsymbol{x})$ as $\widehat{p}_{\mathrm{te}}(\boldsymbol{x}) = \widehat{w}(\boldsymbol{x}) p_{\mathrm{tr}}(\boldsymbol{x})$. In KLIEP, the parameters $\boldsymbol{\alpha}$ are determined so that the Kullback-Leibler divergence from $p_{\mathrm{te}}(\boldsymbol{x})$ to $\widehat{p}_{\mathrm{te}}(\boldsymbol{x})$ is minimized:

$$\mathrm{KL}(\boldsymbol{\alpha}) := \mathop{\mathbb{E}}_{\boldsymbol{x}^{\mathrm{te}}} \left[\log \frac{p_{\mathrm{te}}(\boldsymbol{x}^{\mathrm{te}})}{\widehat{w}(\boldsymbol{x}^{\mathrm{te}}) p_{\mathrm{tr}}(\boldsymbol{x}^{\mathrm{te}})} \right] = \mathop{\mathbb{E}}_{\boldsymbol{x}^{\mathrm{te}}} \left[\log \frac{p_{\mathrm{te}}(\boldsymbol{x}^{\mathrm{te}})}{p_{\mathrm{tr}}(\boldsymbol{x}^{\mathrm{te}})} \right] - \mathop{\mathbb{E}}_{\boldsymbol{x}^{\mathrm{te}}} \left[\log \widehat{w}(\boldsymbol{x}^{\mathrm{te}}) \right],$$

where $\mathbb{E}_{\boldsymbol{x}^{\mathrm{te}}}$ denotes the expectation over $\boldsymbol{x}^{\mathrm{te}}$ drawn from $p_{\mathrm{te}}(\boldsymbol{x})$. The first term is a constant, so it can be safely ignored. We define the negative of the second term by KL$'$:

$$\mathrm{KL}'(\boldsymbol{\alpha}) := \mathop{\mathbb{E}}_{\boldsymbol{x}^{\mathrm{te}}} \left[\log \widehat{w}(\boldsymbol{x}^{\mathrm{te}}) \right]. \tag{6}$$

Since $\widehat{p}_{\mathrm{te}}(\boldsymbol{x})$ $(= \widehat{w}(\boldsymbol{x}) p_{\mathrm{tr}}(\boldsymbol{x}))$ is a probability density function, it should satisfy

$$1 = \int_{\mathcal{D}} \widehat{p}_{\mathrm{te}}(\boldsymbol{x}) d\boldsymbol{x} = \int_{\mathcal{D}} \widehat{w}(\boldsymbol{x}) p_{\mathrm{tr}}(\boldsymbol{x}) d\boldsymbol{x} = \mathop{\mathbb{E}}_{\boldsymbol{x}^{\mathrm{tr}}} \left[\widehat{w}(\boldsymbol{x}^{\mathrm{tr}}) \right]. \tag{7}$$

The KLIEP optimization problem is given by replacing the expectations in Eqs.(6) and (7) with empirical averages:

$$\max_{\{\alpha_\ell\}_{\ell=1}^{n_{\mathrm{te}}}} \left[\sum_{j=1}^{n_{\mathrm{te}}} \log \left(\sum_{\ell=1}^{n_{\mathrm{te}}} \alpha_\ell K(\boldsymbol{x}_j^{\mathrm{te}}, \boldsymbol{x}_\ell^{\mathrm{te}}) \right) \right]$$
$$\text{subject to } \frac{1}{n_{\mathrm{tr}}} \sum_{\ell=1}^{n_{\mathrm{te}}} \alpha_\ell \left(\sum_{i=1}^{n_{\mathrm{tr}}} K(\boldsymbol{x}_i^{\mathrm{tr}}, \boldsymbol{x}_\ell^{\mathrm{te}}) \right) = 1 \text{ and } \alpha_1, \alpha_2, \ldots, \alpha_{n_{\mathrm{te}}} \geq 0.$$

This is a *convex* optimization problem and the global solution—which tends to be *sparse* (Boyd & Vandenberghe, 2004)—can be obtained, e.g., by simply performing gradient ascent and feasibility satisfaction iteratively. A pseudo code is summarized in Figure 8. The Gaussian width σ can be optimized by CV over KL$'$, where only the test samples $\{\boldsymbol{x}_j^{\mathrm{te}}\}_{j=1}^{n_{\mathrm{te}}}$ are divided into k disjoint subsets (Sugiyama et al., 2008).

A MATLAB® implementation of the entire KLIEP algorithm is available from the following web page.

http://sugiyama-www.cs.titech.ac.jp/˜sugi/software/KLIEP/

```
Input: $\{\boldsymbol{x}_i^{\mathrm{tr}}\}_{i=1}^{n_{\mathrm{tr}}}$, $\{\boldsymbol{x}_j^{\mathrm{te}}\}_{j=1}^{n_{\mathrm{te}}}$, and $\sigma$
Output: $\widehat{w}(\boldsymbol{x})$

$A_{j,\ell} \longleftarrow K_\sigma(\boldsymbol{x}_j^{\mathrm{te}}, \boldsymbol{x}_\ell^{\mathrm{te}})$   for $j, \ell = 1, 2, \ldots, n_{\mathrm{te}}$;
$b_\ell \longleftarrow \frac{1}{n_{\mathrm{tr}}} \sum_{i=1}^{n_{\mathrm{tr}}} K_\sigma(\boldsymbol{x}_i^{\mathrm{tr}}, \boldsymbol{x}_\ell^{\mathrm{te}})$   for $\ell = 1, 2, \ldots, n_{\mathrm{te}}$;
Initialize $\boldsymbol{\alpha}$ $(> \mathbf{0}_{n_{\mathrm{te}}})$ and $\varepsilon$ $(0 < \varepsilon \ll 1)$;
Repeat until convergence
    $\boldsymbol{\alpha} \longleftarrow \boldsymbol{\alpha} + \varepsilon \boldsymbol{A}^\top(\mathbf{1}_{n_{\mathrm{te}}} ./ \boldsymbol{A}\boldsymbol{\alpha})$;  % Gradient ascent
    $\boldsymbol{\alpha} \longleftarrow \boldsymbol{\alpha} + (1 - \boldsymbol{b}^\top\boldsymbol{\alpha})\boldsymbol{b}/(\boldsymbol{b}^\top\boldsymbol{b})$;  % Constraint satisfaction
    $\boldsymbol{\alpha} \longleftarrow \max(\mathbf{0}_{n_{\mathrm{te}}}, \boldsymbol{\alpha})$;  % Constraint satisfaction
    $\boldsymbol{\alpha} \longleftarrow \boldsymbol{\alpha}/(\boldsymbol{b}^\top\boldsymbol{\alpha})$;  % Constraint satisfaction
end
$\widehat{w}(\boldsymbol{x}) \longleftarrow \sum_{\ell=1}^{n_{\mathrm{te}}} \alpha_\ell K_\sigma(\boldsymbol{x}, \boldsymbol{x}_\ell^{\mathrm{te}})$;
```

Figure 8: Pseudo code of KLIEP. $\mathbf{0}_{n_{\mathrm{te}}}$ denotes the n_{te}-dimensional vector with all zeros, and $\mathbf{1}_{n_{\mathrm{te}}}$ denotes the n_{te}-dimensional vector with all ones. './' indicates the element-wise division, and inequalities and the 'max' operation for vectors are applied in the element-wise manner.

5.3 Numerical Examples

Here, we illustrate the behavior of the KLIEP method.

Let us consider the following one-dimensional importance estimation problem:

$$p_{\mathrm{tr}}(x) = N(x; 1, (1/2)^2) \ \text{ and } \ p_{\mathrm{te}}(x) = N(x; 2, (1/4)^2).$$

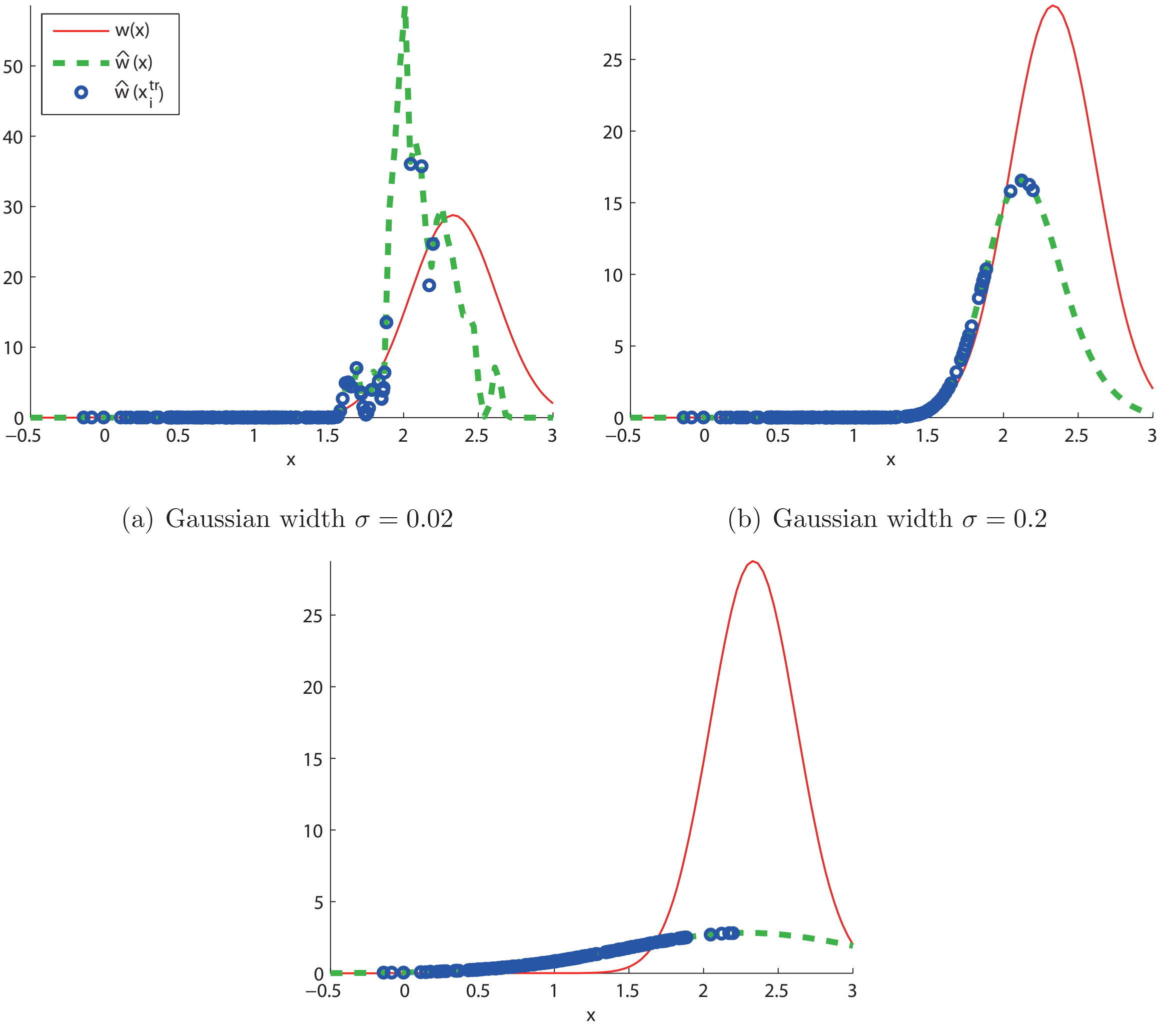
Let the number of training samples be $n_{\mathrm{tr}} = 200$ and the number of test samples be $n_{\mathrm{te}} = 1000$.

Figure 9 depicts the true importance and its estimates by KLIEP, where three different Gaussian widths $\sigma = 0.02, 0.2, 0.8$ are tested. The graphs show that the performance of KLIEP is highly dependent on the Gaussian width. More specifically, the estimated importance function $\widehat{w}(x)$ is highly fluctuated when σ is small, while it is overly smoothed when σ is large. When σ is chosen appropriately, KLIEP seems to work reasonably well for this example.

Figure 10 depicts the values of the true J (see Eq.(6)) and its estimate by 5-fold CV; the means, the 25 percentiles, and the 75 percentiles over 100 trials are plotted as functions of the Gaussian width σ. This shows that CV gives a very good estimate of J, which results in an appropriate choice of σ.

6 Conclusions and Outlook

In standard supervised learning theories, test input points are assumed to follow the same probability distribution as training input points. However, this assumption is often violated in real-world learning problems. In this chapter, we reviewed recently proposed



(a) Gaussian width $\sigma = 0.02$

(b) Gaussian width $\sigma = 0.2$

(c) Gaussian width $\sigma = 0.8$

Figure 9: Results of importance estimation by KLIEP. $w(x)$ is the true importance function and $\widehat{w}(x)$ is its estimation obtained by KLIEP.

techniques for covariate shift adaptation, including importance-weighted empirical risk minimization, importance-weighted cross-validation, and direct importance estimation.

In Section 5, we introduced the KLIEP algorithm for importance estimation, where linearly-parameterized models were used. It was shown that the KLIEP idea can also be naturally applied to log-linear models (Tsuboi et al., 2009), Gaussian mixture models (Yamada & Sugiyama, 2009), and probabilistic principal component analysis mixture models (Yamada et al., 2010b). Other than KLIEP, various methods of direct importance estimation have also been proposed (Silverman, 1978; Ćwik & Mielniczuk, 1989; Qin, 1998; Cheng & Chu, 2004; Huang et al., 2007; Bickel et al., 2007; Kanamori et al., 2009a). Among them, the method proposed in Kanamori et al. (2009a) called *uncon-*

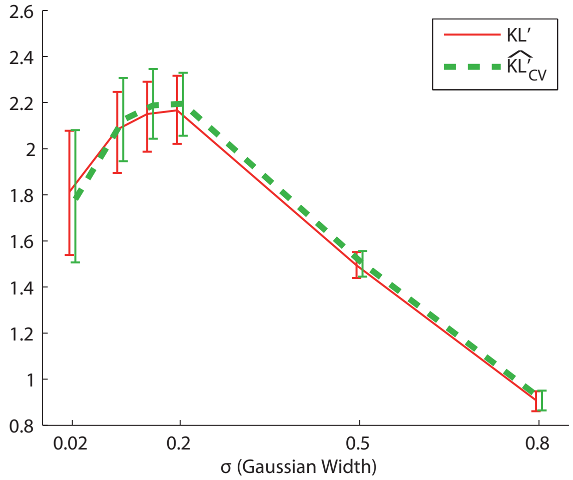


Figure 10: Model selection curve for KLIEP. KL$'$ is the true score of an estimated importance (see Eq.(6)) and $\widehat{\mathrm{KL}'}_{\mathrm{CV}}$ is its estimate by 5-fold CV.

strained least-squares importance fitting (uLSIF) gives an analytic-form solution and the solution can be computed very efficiently in a stable manner. Thus it can be applied to large-scale data sets.

Recently, importance estimation methods which incorporate dimensionality reduction have been developed. A method proposed by Sugiyama et al. (2010a) uses a supervised dimensionality reduction technique called *local Fisher discriminant analysis* (Sugiyama, 2007) for identifying a subspace in which two densities are significantly different (which is called the *hetero-distributional subspace*). Another method proposed by Sugiyama et al. (2011) tries to find the hetero-distributional subspace by directly minimizing the discrepancy between the two distributions. Theoretical analysis of importance estimation has also been conducted thoroughly (Silverman, 1978; Ćwik & Mielniczuk, 1989; Gijbels & Mielniczuk, 1995; Jacob & Oliveira, 1997; Qin, 1998; Cheng & Chu, 2004; Bensaid & Fabre, 2007; Nguyen et al., 2010; Sugiyama et al., 2008; Chen et al., 2009; Kanamori et al., 2009b; Kanamori et al., 2010).

It has been shown that various statistical data processing tasks can be solved through importance estimation (Sugiyama et al., 2009; Sugiyama et al., 2012), including multi-task learning (Bickel et al., 2007), inlier-based outlier detection (Silverman, 1978; Hido et al., 2008; Smola et al., 2009; Hido et al., 2011), change detection in time series (Kawahara & Sugiyama, 2011), mutual information estimation (Suzuki et al., 2008; Suzuki et al., 2009b), independent component analysis (Suzuki & Sugiyama, 2011), feature selection (Suzuki et al., 2009a), sufficient dimension reduction (Suzuki & Sugiyama, 2010), causal inference (Yamada & Sugiyama, 2010), conditional density estimation (Sugiyama et al., 2010b), and probabilistic classification (Sugiyama, 2010). Thus, following this line of research, further improving the accuracy and computational efficiency of importance estimation as well as further exploring possible application of importance estimation would be a promising direction to be pursued.

Acknowledgments

The author was supported by MEXT Grant-in-Aid for Young Scientists (A) 20680007, SCAT, and AOARD.

References

Akaike, H. (1970). Statistical predictor identification. *Annals of the Institute of Statistical Mathematics*, *22*, 203–217.

Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, *AC-19*, 716–723.

Akaike, H. (1980). Likelihood and the Bayes procedure. *Bayesian Statistics* (pp. 141–166). Valencia, Spain: Valencia University Press.

Akiyama, T., Hachiya, H., & Sugiyama, M. (2010). Efficient exploration through active learning for value function approximation in reinforcement learning. *Neural Networks*, *23*, 639–648.

Bensaid, N., & Fabre, J. P. (2007). Optimal asymptotic quadratic error of kernel estimators of Radon-Nikodym derivatives for strong mixing data. *Journal of Nonparametric Statistics*, *19*, 77–88.

Bickel, S., Brückner, M., & Scheffer, T. (2007). Discriminative learning for differing training and test distributions. *Proceedings of the 24th International Conference on Machine Learning (ICML2007)* (pp. 81–88).

Bishop, C. M. (1995). *Neural networks for pattern recognition.* Oxford, UK: Clarendon Press.

Boyd, S., & Vandenberghe, L. (2004). *Convex optimization.* Cambridge, UK: Cambridge University Press.

Breiman, L. (1998). Arcing classifiers. *The Annals of Statistics*, *26*, 801–849.

Chen, S.-M., Hsu, Y.-S., & Liaw, J.-T. (2009). On kernel estimators of density ratio. *Statistics*, *43*, 463–479.

Chen, S. S., Donoho, D. L., & Saunders, M. A. (1998). Atomic decomposition by basis pursuit. *SIAM Journal on Scientific Computing*, *20*, 33–61.

Cheng, K. F., & Chu, C. K. (2004). Semiparametric density estimation under a two-sample density ratio model. *Bernoulli*, *10*, 583–604.

Craven, P., & Wahba, G. (1979). Smoothing noisy data with spline functions: Estimating the correct degree of smoothing by the method of generalized cross-validation. *Numerische Mathematik*, *31*, 377–403.

Ćwik, J., & Mielniczuk, J. (1989). Estimating density ratio with application to discriminant analysis. *Communications in Statistics: Theory and Methods*, *18*, 3057–3069.

Duda, R. O., Hart, P. E., & Stork, D. G. (2001). *Pattern classification*. New York, NY, USA: Wiley. Second edition.

Efron, B., & Tibshirani, R. J. (1993). *An introduction to the bootstrap*. New York, NY, USA: Chapman & Hall/CRC.

Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, *7*, 179–188.

Fishman, G. S. (1996). *Monte Carlo: Concepts, algorithms, and applications*. Berlin, Germany: Springer-Verlag.

Freund, Y., & Schapire, R. E. (1996). Experiments with a new boosting algorithm. *Proc. 13th International Conference on Machine Learning* (pp. 148–156). Morgan Kaufmann.

Friedman, J., Hastie, T., & Tibshirani, R. (2000). Additive logistic regression: A statistical view of boosting. *The Annals of Statistics*, *28*, 337–407.

Gijbels, I., & Mielniczuk, J. (1995). Asymptotic properties of kernel estimators of the Radon-Nikodym derivative with applications to discriminant analysis. *Statistica Sinica*, *5*, 261–278.

Hachiya, H., Akiyama, T., Sugiyama, M., & Peters, J. (2009). Adaptive importance sampling for value function approximation in off-policy reinforcement learning. *Neural Networks*, *22*, 1399–1410.

Hachiya, H., Peters, J., & Sugiyama, M. (2011). Reward weighted regression with sample reuse. *Neural Computation*, *11*, 2798–2832.

Härdle, W., Müller, M., Sperlich, S., & Werwatz, A. (2004). *Nonparametric and semiparametric models*. Berlin, Germany: Springer.

Hastie, T., Tibshirani, R., & Friedman, J. (2001). *The elements of statistical learning: Data mining, inference, and prediction*. New York, NY, USA: Springer.

Henkel, R. E. (1976). *Tests of significance*. Beverly Hills, CA, USA.: SAGE Publication.

Hido, S., Tsuboi, Y., Kashima, H., Sugiyama, M., & Kanamori, T. (2008). Inlier-based outlier detection via direct density ratio estimation. *Proceedings of IEEE International Conference on Data Mining (ICDM2008)* (pp. 223–232). Pisa, Italy.

Hido, S., Tsuboi, Y., Kashima, H., Sugiyama, M., & Kanamori, T. (2011). Statistical outlier detection using direct density ratio estimation. *Knowledge and Information Systems*, *26*, 309–336.

Huang, J., Smola, A., Gretton, A., Borgwardt, K. M., & Schölkopf, B. (2007). Correcting sample selection bias by unlabeled data. In B. Schölkopf, J. Platt and T. Hoffman (Eds.), *Advances in neural information processing systems 19*, 601–608. Cambridge, MA, USA: MIT Press.

Ishiguro, M., Sakamoto, Y., & Kitagawa, G. (1997). Bootstrapping log likelihood and EIC, an extension of AIC. *Annals of the Institute of Statistical Mathematics*, *49*, 411–434.

Jacob, P., & Oliveira, P. E. (1997). Kernel estimators of general Radon-Nikodym derivatives. *Statistics*, *30*, 25–46.

Kanamori, T. (2007). Pool-based active learning with optimal sampling distribution and its information geometrical interpretation. *Neurocomputing*, *71*, 353–362.

Kanamori, T., Hido, S., & Sugiyama, M. (2009a). A least-squares approach to direct importance estimation. *Journal of Machine Learning Research*, *10*, 1391–1445.

Kanamori, T., & Shimodaira, H. (2003). Active learning algorithm using the maximum weighted log-likelihood estimator. *Journal of Statistical Planning and Inference*, *116*, 149–162.

Kanamori, T., Suzuki, T., & Sugiyama, M. (2009b). *Condition number analysis of kernel-based density ratio estimation* (Technical Report). arXiv.

Kanamori, T., Suzuki, T., & Sugiyama, M. (2010). Theoretical analysis of density ratio estimation. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, *E93-A*, 787–798.

Kawahara, Y., & Sugiyama, M. (2011). Sequential change-point detection based on direct density-ratio estimation. *Statistical Analysis and Data Mining*. to appear.

Konishi, S., & Kitagawa, G. (1996). Generalized information criteria in model selection. *Biometrika*, *83*, 875–890.

Kullback, S., & Leibler, R. A. (1951). On information and sufficiency. *Annals of Mathematical Statistics*, *22*, 79–86.

Luntz, A., & Brailovsky, V. (1969). On estimation of characters obtained in statistical procedure of recognition. *Technicheskaya Kibernetica*, *3*. in Russian.

Mallows, C. L. (1973). Some comments on C_P. *Technometrics*, *15*, 661–675.

Mangasarian, O. L., & Musicant, D. R. (2000). Robust linear and support vector regression. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *22*, 950–955.

Murata, N., Yoshizawa, S., & Amari, S. (1994). Network information criterion — Determining the number of hidden units for an artificial neural network model. *IEEE Transactions on Neural Networks*, *5*, 865–872.

Nguyen, X., Wainwright, M. J., & Jordan, M. I. (2010). Estimating divergence functionals and the likelihood ratio by convex risk minimization. *IEEE Transactions on Information Theory*, *56*, 5847–5861.

Qin, J. (1998). Inferences for case-control and semiparametric two-sample density ratio models. *Biometrika*, *85*, 619–630.

Quiñonero-Candela, J., Sugiyama, M., Schwaighofer, A., & Lawrence, N. (Eds.). (2009). *Dataset shift in machine learning*. Cambridge, MA, USA: MIT Press.

Rissanen, J. (1978). Modeling by shortest data description. *Automatica*, *14*, 465–471.

Rissanen, J. (1987). Stochastic complexity. *Journal of the Royal Statistical Society, Series B*, *49*, 223–239.

Schölkopf, B., & Smola, A. J. (2002). *Learning with kernels*. Cambridge, MA, USA: MIT Press.

Schwarz, G. (1978). Estimating the dimension of a model. *The Annals of Statistics*, *6*, 461–464.

Shibata, R. (1989). Statistical aspects of model selection. In J. C. Willems (Ed.), *From data to model*, 215–240. New York, NY, USA: Springer-Verlag.

Shimodaira, H. (2000). Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, *90*, 227–244.

Silverman, B. W. (1978). Density ratios, empirical likelihood and cot death. *Journal of the Royal Statistical Society, Series C*, *27*, 26–33.

Smola, A., Song, L., & Teo, C. H. (2009). Relative novelty detection. *Proceedings of Twelfth International Conference on Artificial Intelligence and Statistics (AISTATS2009)* (pp. 536–543). Clearwater Beach, FL, USA.

Stone, M. (1974). Cross-validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society, Series B*, *36*, 111–147.

Sugiyama, M. (2006). Active learning in approximately linear regression based on conditional expectation of generalization error. *Journal of Machine Learning Research*, *7*, 141–166.

Sugiyama, M. (2007). Dimensionality reduction of multimodal labeled data by local Fisher discriminant analysis. *Journal of Machine Learning Research*, *8*, 1027–1061.

Sugiyama, M. (2010). Superfast-trainable multi-class probabilistic classifier by least-squares posterior fitting. *IEICE Transactions on Information and Systems*, *E93-D*, 2690–2701.

Sugiyama, M., Kanamori, T., Suzuki, T., Hido, S., Sese, J., Takeuchi, I., & Wang, L. (2009). A density-ratio framework for statistical data processing. *IPSJ Transactions on Computer Vision and Applications*, *1*, 183–208.

Sugiyama, M., & Kawanabe, M. (2011). *Machine learning in non-stationary environments: Introduction to covariate shift adaptation*. Cambridge, MA, USA: MIT Press. to appear.

Sugiyama, M., Kawanabe, M., & Chui, P. L. (2010a). Dimensionality reduction for density ratio estimation in high-dimensional spaces. *Neural Networks*, *23*, 44–59.

Sugiyama, M., Kawanabe, M., & Müller, K.-R. (2004). Trading variance reduction with unbiasedness: The regularized subspace information criterion for robust model selection in kernel regression. *Neural Computation*, *16*, 1077–1104.

Sugiyama, M., Krauledat, M., & Müller, K.-R. (2007). Covariate shift adaptation by importance weighted cross validation. *Journal of Machine Learning Research*, *8*, 985–1005.

Sugiyama, M., & Müller, K.-R. (2002). The subspace information criterion for infinite dimensional hypothesis spaces. *Journal of Machine Learning Research*, *3*, 323–359.

Sugiyama, M., & Müller, K.-R. (2005). Input-dependent estimation of generalization error under covariate shift. *Statistics & Decisions*, *23*, 249–279.

Sugiyama, M., & Nakajima, S. (2009). Pool-based active learning in approximate linear regression. *Machine Learning*, *75*, 249–274.

Sugiyama, M., & Ogawa, H. (2001). Subspace information criterion for model selection. *Neural Computation*, *13*, 1863–1889.

Sugiyama, M., Suzuki, T., & Kanamori, T. (2012). *Density ratio estimation in machine learning*. Cambridge, UK: Cambridge University Press. to appear.

Sugiyama, M., Suzuki, T., Nakajima, S., Kashima, H., von Bünau, P., & Kawanabe, M. (2008). Direct importance estimation for covariate shift adaptation. *Annals of the Institute of Statistical Mathematics*, *60*, 699–746.

Sugiyama, M., Takeuchi, I., Suzuki, T., Kanamori, T., Hachiya, H., & Okanohara, D. (2010b). Least-squares conditional density estimation. *IEICE Transactions on Information and Systems*, *E93-D*, 583–594.

Sugiyama, M., Yamada, M., von Bünau, P., Suzuki, T., Kanamori, T., & Kawanabe, M. (2011). Direct density-ratio estimation with dimensionality reduction via least-squares hetero-distributional subspace search. *Neural Networks*, *24*, 183–198.

Suzuki, T., & Sugiyama, M. (2010). Sufficient dimension reduction via squared-loss mutual information estimation. *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics (AISTATS2010)* (pp. 804–811). Sardinia, Italy.

Suzuki, T., & Sugiyama, M. (2011). Least-squares independent component analysis. *Neural Computation*, *23*, 284–301.

Suzuki, T., Sugiyama, M., Kanamori, T., & Sese, J. (2009a). Mutual information estimation reveals global associations between stimuli and biological processes. *BMC Bioinformatics*, *10*, S52.

Suzuki, T., Sugiyama, M., Sese, J., & Kanamori, T. (2008). Approximating mutual information by maximum likelihood density ratio estimation. *Proceedings of ECML-PKDD2008 Workshop on New Challenges for Feature Selection in Data Mining and Knowledge Discovery 2008 (FSDM2008)* (pp. 5–20). Antwerp, Belgium.

Suzuki, T., Sugiyama, M., & Tanaka, T. (2009b). Mutual information approximation via maximum likelihood estimation of density ratio. *Proceedings of 2009 IEEE International Symposium on Information Theory (ISIT2009)* (pp. 463–467). Seoul, Korea.

Takeuchi, K. (1976). Distribution of information statistics and validity criteria of models. *Mathematical Science*, *153*, 12–18. in Japanese.

Tibshirani, R. (1996). Regression shrinkage and subset selection with the lasso. *Journal of the Royal Statistical Society, Series B*, *58*, 267–288.

Tsuboi, Y., Kashima, H., Hido, S., Bickel, S., & Sugiyama, M. (2009). Direct density ratio estimation for large-scale covariate shift adaptation. *Journal of Information Processing*, *17*, 138–155.

Ueki, K., Sugiyama, M., & Ihara, Y. (2011). Lighting condition adaptation for perceived age estimation. *IEICE Transactions on Information and Systems*, *E94-D*, 392–395.

Vapnik, V. N. (1998). *Statistical learning theory.* New York, NY, USA: Wiley.

Wahba, G. (1990). *Spline models for observational data.* Philadelphia, PA, USA: Society for Industrial and Applied Mathematics.

Watanabe, S. (2009). *Algebraic geometry and statistical learning theory.* Cambridge, UK: Cambridge University Press.

Wiens, D. P. (2000). Robust weights and designs for biased regression models: Least squares and generalized M-estimation. *Journal of Statistical Planning and Inference*, *83*, 395–412.

Williams, P. M. (1995). Bayesian regularization and pruning using a Laplace prior. *Neural Computation*, *7*, 117–143.

Yamada, M., & Sugiyama, M. (2009). Direct importance estimation with Gaussian mixture models. *IEICE Transactions on Information and Systems*, *E92-D*, 2159–2162.

Yamada, M., & Sugiyama, M. (2010). Dependence minimizing regression with model selection for non-linear causal inference under non-Gaussian noise. *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence (AAAI2010)* (pp. 643–648). Atlanta, Georgia, USA: The AAAI Press.

Yamada, M., Sugiyama, M., & Matsui, T. (2010a). Semi-supervised speaker identification under covariate shift. *Signal Processing*, *90*, 2353–2361.

Yamada, M., Sugiyama, M., Wichern, G., & Simm, J. (2010b). Direct importance estimation with a mixture of probabilistic principal component analyzers. *IEICE Transactions on Information and Systems*, *E93-D*, 2846–2849.

Zadrozny, B. (2004). Learning and evaluating classifiers under sample selection bias. *Proceedings of the Twenty-First International Conference on Machine Learning (ICML2004)* (pp. 903–910). New York, NY, USA: ACM Press.

Cross-Domain Object Matching with Model Selection

Makoto Yamada
Tokyo Institute of Technology
yamada@sg.cs.titech.ac.jp

Masashi Sugiyama
Tokyo Institute of Technology
sugi@cs.titech.ac.jp

Abstract

The goal of *cross-domain object matching* (CDOM) is to find correspondence between two sets of objects in different domains in an unsupervised way. Photo album summarization is a typical application of CDOM, where photos are automatically aligned into a designed frame expressed in the Cartesian coordinate system. CDOM is usually formulated as finding a mapping from objects in one domain (photos) to objects in the other domain (frame) so that the pairwise dependency is maximized. A state-of-the-art CDOM method employs a kernel-based dependency measure, but it has a drawback that the kernel parameter needs to be determined manually. In this paper, we propose alternative CDOM methods that can naturally address the model selection problem. Through experiments on image matching, unpaired voice conversion, and photo album summarization tasks, the effectiveness of the proposed methods is demonstrated.

1 Introduction

The objective of *cross-domain object matching* (CDOM) is to match two sets of objects in different domains. For instance, in photo album summarization, photos are automatically assigned into a designed frame expressed in the Cartesian coordinate system. A typical approach of CDOM is to find a mapping from objects in one domain (photos) to objects in the other domain (frame) so that the pairwise dependency is maximized. In this scenario, accurately evaluating the dependence between objects is a key challenge.

Kernelized sorting (KS) (Jebara, 2004) tries to find a mapping between two domains that maximizes the *mutual information* (MI) (Cover and Thomas, 2006) under the Gaussian assumption. However, since the Gaussian assumption may not be fulfilled in practice, this method (which we refer to as KS-MI) tends to perform poorly.

To overcome the limitation of KS-MI, Quadrianto *et al.* (2010) proposed using the kernel-based dependence measure called the *Hilbert-Schmidt independence criterion* (HSIC) (Gretton *et al.*, 2005) for KS. Since HSIC is distribution-free, KS with HSIC (which we refer to as KS-HSIC) is more flexible than KS-MI. However, HSIC includes a tuning parameter (more specifically, the Gaussian kernel width), and its choice is crucial to obtain better performance (see also Jagarlamudi *et al.*, 2010). Although using the median distance between sample points as the Gaussian kernel width is a common heuristic in kernel-based dependence measures (see e.g., Fukumizu *et al.*, 2009a), this does not always perform well in practice.

In this paper, we propose two alternative CDOM methods that can naturally address the model selection problem. The first method employs another kernel-based dependence measure based on the *normalized cross-covariance operator* (NOCCO) (Fukumizu *et al.*, 2009b), which we refer to as KS-NOCCO. The NOCCO-based dependence measure was shown to be asymptotically independent of the choice of kernels. Thus, KS-NOCCO is expected to be less sensitive to the kernel parameter choice, which is an advantage over HSIC.

The second method uses *least-squares mutual information* (LSMI) (Suzuki *et al.*, 2009) as the dependence measure, which is a consistent estimator of the *squared-loss mutual information* (SMI) achieving the optimal convergence rate. We call this method *least-squares object matching* (LSOM). An advantage of LSOM is that cross-validation (CV) with respect to the LSMI criterion is possible. Thus, all the tuning parameters such as the Gaussian kernel width and the regularization parameter can be objectively determined by

Appearing in Proceedings of the 14^{th} International Conference on Artificial Intelligence and Statistics (AISTATS) 2011, Fort Lauderdale, FL, USA. Volume 15 of JMLR: W&CP 15.

CV.

Through experiments on image matching, unpaired voice conversion, and photo album summarization tasks, LSOM is shown to be the most promising approach to CDOM.

2 Problem Formulation

In this section, we formulate the problem of *cross-domain object matching* (CDOM).

The goal of CDOM is, given two sets of samples of the same size, $\{\boldsymbol{x}_i\}_{i=1}^n$ and $\{\boldsymbol{y}_i\}_{i=1}^n$, to find a mapping that well "matches" them.

Let π be a permutation function over $\{1,\ldots,n\}$, and let $\boldsymbol{\Pi}$ be the corresponding permutation indicator matrix, i.e.,

$$\boldsymbol{\Pi} \in \{0,1\}^{n\times n},\ \boldsymbol{\Pi}\boldsymbol{1}_n = \boldsymbol{1}_n,\ \text{and}\ \boldsymbol{\Pi}^\top\boldsymbol{1}_n = \boldsymbol{1}_n,$$

where $\boldsymbol{1}_n$ is the n-dimensional vector with all ones and $^\top$ denotes the transpose. Let us denote the samples matched by a permutation π by

$$Z(\boldsymbol{\Pi}) := \{(\boldsymbol{x}_i, \boldsymbol{y}_{\pi(i)})\}_{i=1}^n.$$

The optimal permutation, denoted by $\boldsymbol{\Pi}^*$, can be obtained as the maximizer of the dependency between the two sets $\{\boldsymbol{x}_i\}_{i=1}^n$ and $\{\boldsymbol{y}_i\}_{i=1}^n$:

$$\boldsymbol{\Pi}^* := \underset{\boldsymbol{\Pi}}{\operatorname{argmax}}\, D(Z(\boldsymbol{\Pi})),$$

where D is some dependence measure.

3 Existing Methods

In this section, we review two existing methods for CDOM, and point out their potential weaknesses.

3.1 Kernelized Sorting with Mutual Information

Kernelized sorting with mutual information (KS-MI) (Jebara, 2004) matches objects in different domains so that MI between matched pairs is maximized. Here, we review KS-MI following alternative derivation provided in Quadrianto *et al.* (2010).

MI is one of the popular dependence measures between random variables. For random variables X and Y, MI is defined as follows (Cover and Thomas, 2006):

$$\mathrm{MI}(Z) := \iint p(\boldsymbol{x},\boldsymbol{y}) \log \frac{p(\boldsymbol{x},\boldsymbol{y})}{p(\boldsymbol{x})p(\boldsymbol{y})} \mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{y},$$

where $p(\boldsymbol{x},\boldsymbol{y})$ denotes the joint density of $\boldsymbol{x}$ and $\boldsymbol{y}$, and $p(\boldsymbol{x})$ and $p(\boldsymbol{y})$ are marginal densities of $\boldsymbol{x}$ and $\boldsymbol{y}$, respectively. MI is zero if and only if $\boldsymbol{x}$ and $\boldsymbol{y}$ are independent, and thus it may be used as a dependency measure. Let $H(X)$, $H(Y)$, and $H(X,Y)$ be the entropies of X and Y and the joint entropy of X and Y, respectively:

$$H(X) = -\int p(\boldsymbol{x}) \log p(\boldsymbol{x}) \mathrm{d}\boldsymbol{x},$$

$$H(Y) = -\int p(\boldsymbol{y}) \log p(\boldsymbol{y}) \mathrm{d}\boldsymbol{y},$$

$$H(X,Y) = -\iint p(\boldsymbol{x},\boldsymbol{y}) \log p(\boldsymbol{x},\boldsymbol{y}) \mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{y}.$$

Then MI between X and Y can be written as

$$\mathrm{MI}(Z) = H(X) + H(Y) - H(X,Y).$$

Since $H(X)$ and $H(Y)$ are independent of permutation $\boldsymbol{\Pi}$, maximizing MI is equivalent to minimizing the joint entropy $H(X,Y)$. If $p(\boldsymbol{x},\boldsymbol{y})$ is Gaussian with covariance matrix $\boldsymbol{\Sigma}$, the joint entropy is expressed as

$$H(X,Y) = \frac{1}{2}\log|\boldsymbol{\Sigma}| + \mathrm{Const.},$$

where $|\boldsymbol{\Sigma}|$ denotes the determinant of matrix $\boldsymbol{\Sigma}$.

Now, let us assume that $\boldsymbol{x}$ and $\boldsymbol{y}$ are jointly normal in some reproducing Kernel Hilbert Spaces (RKHSs) endowed with joint kernel $K(\boldsymbol{x},\boldsymbol{x}')L(\boldsymbol{y},\boldsymbol{y}')$, where $K(\boldsymbol{x},\boldsymbol{x}')$ and $L(\boldsymbol{y},\boldsymbol{y}')$ are reproducing kernels for $\boldsymbol{x}$ and $\boldsymbol{y}$, respectively. Then KS-MI is formulated as follows:

$$\min_{\boldsymbol{\Pi}} \log|\boldsymbol{\Gamma}(\boldsymbol{K}\circ(\boldsymbol{\Pi}^\top\boldsymbol{L}\boldsymbol{\Pi}))\boldsymbol{\Gamma}|, \tag{1}$$

where $\boldsymbol{K} = \{K(\boldsymbol{x}_i,\boldsymbol{x}_j)\}_{i,j=1}^n$ and $\boldsymbol{L} = \{L(\boldsymbol{y}_i,\boldsymbol{y}_j)\}_{i,j=1}^n$ are kernel matrices, $\circ$ denotes the Hadamard product (a.k.a. the element-wise product), $\boldsymbol{\Gamma} = \boldsymbol{I}_n - \frac{1}{n}\boldsymbol{1}_n\boldsymbol{1}_n^\top$ is the centering matrix, and $\boldsymbol{I}_n$ is the n-dimensional identity matrix.

A critical weakness of KS-MI is the Gaussian assumption, which may not be fulfilled in practice.

3.2 Kernelized Sorting with Hilbert-Schmidt Independence Criterion

Kernelized sorting with Hilbert-Schmidt independence criterion (KS-HSIC) matches objects in different domains so that HSIC between matched pairs is maximized.

HSIC is a kernel-based dependence measure given as follows (Gretton *et al.*, 2005):

$$\mathrm{HSIC}(Z) = \mathrm{tr}(\bar{\boldsymbol{K}}\bar{\boldsymbol{L}}),$$

where $\bar{\boldsymbol{K}} = \boldsymbol{\Gamma}\boldsymbol{K}\boldsymbol{\Gamma}$ and $\bar{\boldsymbol{L}} = \boldsymbol{\Gamma}\boldsymbol{L}\boldsymbol{\Gamma}$ are the centered kernel matrices for $\boldsymbol{x}$ and $\boldsymbol{y}$, respectively. Note that smaller HSIC scores mean that X and Y are closer to be independent.

KS-HSIC is formulated as follows (Quadrianto *et al.*, 2010):

$$\max_{\boldsymbol{\Pi}} \mathrm{HSIC}(Z(\boldsymbol{\Pi})), \tag{2}$$

where

$$\mathrm{HSIC}(Z(\boldsymbol{\Pi})) = \mathrm{tr}(\bar{\boldsymbol{K}}\boldsymbol{\Pi}^\top \bar{\boldsymbol{L}}\boldsymbol{\Pi}). \tag{3}$$

This optimization problem is called the *quadratic assignment problem* (QAP) (Finke *et al.*, 1987), and it is known to be *NP-hard*. There exists several QAP solvers based on, e.g., simulated annealing, tabu search, and genetic algorithms. However, those QAP solvers are not easy to use in practice since they contain various tuning parameters.

Another approach to solving Eq.(2) based on a *linear assignment problem* (LAP) (Kuhn, 1955) was proposed in Quadrianto *et al.* (2010), which is explained below. Let us relax the permutation indicator matrix $\boldsymbol{\Pi}$ to take real values:

$$\boldsymbol{\Pi} \in [0,1]^{n\times n},\ \boldsymbol{\Pi}\boldsymbol{1}_n = \boldsymbol{1}_n,\ \text{and}\ \boldsymbol{\Pi}^\top \boldsymbol{1}_n = \boldsymbol{1}_n. \tag{4}$$

Then, Eq.(3) is convex with respect to $\boldsymbol{\Pi}$ (see Lemma 7 in Quadrianto *et al.*, 2010), and its lower bound can be obtained using some $\widetilde{\boldsymbol{\Pi}}$ as follows:

$$\begin{aligned}
&\mathrm{tr}(\bar{\boldsymbol{K}}\boldsymbol{\Pi}^\top \bar{\boldsymbol{L}}\boldsymbol{\Pi}) \\
&\geq \mathrm{tr}(\bar{\boldsymbol{K}}\widetilde{\boldsymbol{\Pi}}^\top \bar{\boldsymbol{L}}\widetilde{\boldsymbol{\Pi}}) + \langle \boldsymbol{\Pi} - \widetilde{\boldsymbol{\Pi}}, \frac{\partial \mathrm{HSIC}(Z(\widetilde{\boldsymbol{\Pi}}))}{\partial \boldsymbol{\Pi}} \rangle \\
&= 2\mathrm{tr}(\bar{\boldsymbol{K}}\boldsymbol{\Pi}^\top \bar{\boldsymbol{L}}\widetilde{\boldsymbol{\Pi}}) - \mathrm{tr}(\bar{\boldsymbol{K}}\widetilde{\boldsymbol{\Pi}}^\top \bar{\boldsymbol{L}}\widetilde{\boldsymbol{\Pi}}),
\end{aligned}$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product between matrices. Based on the above lower bound, Quadrianto *et al.* (2010) proposed to update the permutation matrix as

$$\boldsymbol{\Pi}^{\text{new}} = (1-\eta)\boldsymbol{\Pi}^{\text{old}} + \eta \operatorname*{argmax}_{\boldsymbol{\Pi}} \mathrm{tr}\left(\boldsymbol{\Pi}^\top \bar{\boldsymbol{L}}\boldsymbol{\Pi}^{\text{old}}\bar{\boldsymbol{K}}\right), \tag{5}$$

where $0 < \eta \leq 1$ is a step size. The second term is an LAP subproblem, which can be efficiently solved by using the *Hungarian method* (Kuhn, 1955).

In the original KS-HSIC paper (Quadrianto *et al.*, 2010), a C++ implementation of the Hungarian method provided by Cooper[1] was used for solving Eq.(5); then $\boldsymbol{\Pi}$ is kept updated by Eq.(5) until convergence.

In this iterative optimization procedure, the choice of initial permutation matrices is critical to obtain a good solution. Quadrianto *et al.* (2010) proposed the following initialization scheme. Suppose the kernel matrices $\bar{\boldsymbol{K}}$ and $\bar{\boldsymbol{L}}$ are rank one, i.e., for some $\boldsymbol{f}$ and $\boldsymbol{g}$, $\bar{\boldsymbol{K}}$ and $\bar{\boldsymbol{L}}$ can be expressed as $\bar{\boldsymbol{K}} = \boldsymbol{f}\boldsymbol{f}^\top$ and $\bar{\boldsymbol{L}} = \boldsymbol{g}\boldsymbol{g}^\top$. Then HSIC can be written as

$$\mathrm{HSIC}(Z(\boldsymbol{\Pi})) = \|\boldsymbol{f}^\top \boldsymbol{\Pi}\boldsymbol{g}\|^2. \tag{6}$$

The initial permutation matrix is determined so that Eq.(6) is maximized. According to Theorems 368 and 369 in Hardy *et al.* (1952), the maximum of Eq.(6) is attained when the elements of $\boldsymbol{f}$ and $\boldsymbol{\Pi}\boldsymbol{g}$ are ordered in the same way. That is, if the elements of $\boldsymbol{f}$ are ordered in the ascending manner (i.e., $f_1 \leq f_2 \leq \cdots \leq f_n$), the maximum of Eq.(6) is attained by ordering the elements of $\boldsymbol{g}$ in the same ascending way. However, since the kernel matrices $\bar{\boldsymbol{K}}$ and $\bar{\boldsymbol{L}}$ may not be rank one in practice, the principal eigenvectors of $\bar{\boldsymbol{K}}$ and $\bar{\boldsymbol{L}}$ were used as $\boldsymbol{f}$ and $\boldsymbol{g}$ in the original KS-HSIC paper (Quadrianto *et al.*, 2010). We call this *eigenvalue-based initialization.*

Since HSIC is a distribution-free dependence measure, KS-HSIC is more flexible than KS-MI. However, a critical weakness of HSIC is that its performance is sensitive to the choice of kernels (Jagarlamudi *et al.*, 2010). A practical heuristic is to use the Gaussian kernel with width set to the median distance between samples (see e.g., Fukumizu *et al.*, 2009a), but this does not always work well in practice.

4 Proposed Methods

In this section, we propose two alternative CDOM methods that can naturally address the model selection problem.

4.1 Kernelized Sorting with Normalized Cross-Covariance Operator

The kernel-based dependence measure based on the *normalized cross-covariance operator* (NOCCO) (Fukumizu *et al.*, 2009b) is given as follows (Fukumizu *et al.*, 2009b):

$$\mathrm{D}_{\mathrm{NOCCO}}(Z) = \mathrm{tr}(\widetilde{\boldsymbol{K}}\widetilde{\boldsymbol{L}}),$$

where $\widetilde{\boldsymbol{K}} = \bar{\boldsymbol{K}}(\bar{\boldsymbol{K}} + n\epsilon\boldsymbol{I}_n)^{-1}$, $\widetilde{\boldsymbol{L}} = \bar{\boldsymbol{L}}(\bar{\boldsymbol{L}} + n\epsilon\boldsymbol{I}_n)^{-1}$, and $\epsilon > 0$ is a regularization parameter. $\mathrm{D}_{\mathrm{NOCCO}}$ was shown to be asymptotically independent of the choice of kernels. Thus, KS with $\mathrm{D}_{\mathrm{NOCCO}}$ (KS-NOCCO) is expected to be less sensitive to the kernel parameter choice than KS-HSIC.

[1] http://mit.edu/harold/www/code.html

The permuted version of $\widetilde{\boldsymbol{L}}$ can be written as

$$\begin{aligned}\widetilde{\boldsymbol{L}}(\boldsymbol{\Pi}) &= \boldsymbol{\Pi}^\top \bar{\boldsymbol{L}}\boldsymbol{\Pi}(\boldsymbol{\Pi}^\top \bar{\boldsymbol{L}}\boldsymbol{\Pi} + n\epsilon \boldsymbol{I}_n)^{-1}\\ &= \boldsymbol{\Pi}^\top \bar{\boldsymbol{L}}(\bar{\boldsymbol{L}} + n\epsilon \boldsymbol{I}_n)^{-1}\boldsymbol{\Pi}\\ &= \boldsymbol{\Pi}^\top \widetilde{\boldsymbol{L}}\boldsymbol{\Pi},\end{aligned}$$

where we used the orthogonality of $\boldsymbol{\Pi}$ (i.e., $\boldsymbol{\Pi}^\top\boldsymbol{\Pi} = \boldsymbol{\Pi}\boldsymbol{\Pi}^\top = \boldsymbol{I}_n$). Thus, the dependency measure for $Z(\boldsymbol{\Pi})$ can be written as

$$\mathrm{D}_{\mathrm{NOCCO}}(Z(\boldsymbol{\Pi})) = \mathrm{tr}(\widetilde{\boldsymbol{K}}\boldsymbol{\Pi}^\top \widetilde{\boldsymbol{L}}\boldsymbol{\Pi}).$$

Since this is essentially the same form as HSIC, a local optimal solution may be obtained in the same way as KS-HSIC:

$$\boldsymbol{\Pi}^{\mathrm{new}} = (1-\eta)\boldsymbol{\Pi}^{\mathrm{old}} + \eta \operatorname*{argmax}_{\boldsymbol{\Pi}} \mathrm{tr}\left(\boldsymbol{\Pi}^\top \widetilde{\boldsymbol{L}}\boldsymbol{\Pi}^{\mathrm{old}}\widetilde{\boldsymbol{K}}\right). \tag{7}$$

However, the property that $\mathrm{D}_{\mathrm{NOCCO}}$ is independent of the kernel choice holds only asymptotically. Thus, with finite samples, $\mathrm{D}_{\mathrm{NOCCO}}$ does still depend on the choice of kernels as well as the regularization parameter ϵ which needs to be manually tuned.

4.2 Least-Squares Object Matching

Next, we propose an alternative method called *least-squares object matching* (LSOM), in which we employ *least-squares mutual information* (LSMI) (Suzuki *et al.*, 2009) as a dependency measure. LSMI is a consistent estimator of the *squared-loss mutual information* (SMI) with the optimal convergence rate. SMI is defined and expressed as

$$\begin{aligned}&\mathrm{SMI}(Z)\\ &= \frac{1}{2}\iint \left(\frac{p(\boldsymbol{x},\boldsymbol{y})}{p(\boldsymbol{x})p(\boldsymbol{y})} - 1\right)^2 p(\boldsymbol{x})p(\boldsymbol{y})\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{y}\\ &= \frac{1}{2}\iint \left(\frac{p(\boldsymbol{x},\boldsymbol{y})}{p(\boldsymbol{x})p(\boldsymbol{y})}\right) p(\boldsymbol{x},\boldsymbol{y})\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{y} - \frac{1}{2}. \end{aligned} \tag{8}$$

Note that SMI is the *Pearson divergence* (Pearson, 1900) from $p(\boldsymbol{x},\boldsymbol{y})$ to $p(\boldsymbol{x})p(\boldsymbol{y})$, while the ordinary MI is the *Kullback-Leibler divergence* (Kullback and Leibler, 1951) from $p(\boldsymbol{x},\boldsymbol{y})$ to $p(\boldsymbol{x})p(\boldsymbol{y})$. SMI is zero if and only if $\boldsymbol{x}$ and $\boldsymbol{y}$ are independent, as the ordinary MI. Its estimator LSMI is given as follows (Suzuki *et al.*, 2009) (see Appendix for the derivation of LSMI):

$$\mathrm{LSMI}(Z) = \frac{1}{2}\widehat{\boldsymbol{\alpha}}^\top \widehat{\boldsymbol{h}} - \frac{1}{2},$$

where

$$\begin{aligned}\widehat{\boldsymbol{\alpha}} &= (\widehat{\boldsymbol{H}} + \lambda \boldsymbol{I}_n)^{-1}\widehat{\boldsymbol{h}},\\ \widehat{\boldsymbol{H}} &= \frac{1}{n^2}(\boldsymbol{K}\boldsymbol{K}^\top) \circ (\boldsymbol{L}\boldsymbol{L}^\top),\\ \widehat{\boldsymbol{h}} &= \frac{1}{n}(\boldsymbol{K}\circ\boldsymbol{L})\boldsymbol{1}_n.\end{aligned}$$

Here, λ (≥ 0) is the regularization parameter. Since cross-validation (CV) with respect to SMI is possible for model selection, tuning parameters in LSMI (i.e., the kernel parameters and the regularization parameter) can be objectively optimized. This is a notable advantage over kernel-based approaches such as KS-HSIC and KS-NOCCO, since the choice of kernels heavily affects the sensitivity of the independence measure in the kernel-based independence measures (Fukumizu *et al.*, 2009a).

Below, we use the following equivalent expression of LSMI:

$$\mathrm{LSMI}(Z) = \frac{1}{2n}\mathrm{tr}\left(\boldsymbol{L}\widehat{\boldsymbol{A}}\boldsymbol{K}\right) - \frac{1}{2}, \tag{9}$$

where $\widehat{\boldsymbol{A}}$ is the diagonal matrix with diagonal elements given by $\widehat{\boldsymbol{\alpha}}$. Note that we used Eq.(73) and Eq.(75) in Minka (2000) for obtaining the above expression.

LSMI for the permuted data $Z(\boldsymbol{\Pi})$ is given by

$$\mathrm{LSMI}(Z(\boldsymbol{\Pi})) = \frac{1}{2n}\mathrm{tr}\left(\boldsymbol{\Pi}^\top\boldsymbol{L}\boldsymbol{\Pi}\widehat{\boldsymbol{A}}_{\boldsymbol{\Pi}}\boldsymbol{K}\right) - \frac{1}{2},$$

where $\widehat{\boldsymbol{A}}_{\boldsymbol{\Pi}}$ is the diagonal matrix with diagonal elements given by $\widehat{\boldsymbol{\alpha}}_{\boldsymbol{\Pi}}$, and $\widehat{\boldsymbol{\alpha}}_{\boldsymbol{\Pi}}$ is given by

$$\begin{aligned}\widehat{\boldsymbol{\alpha}}_{\boldsymbol{\Pi}} &= (\widehat{\boldsymbol{H}}_{\boldsymbol{\Pi}} + \lambda \boldsymbol{I}_n)^{-1}\widehat{\boldsymbol{h}}_{\boldsymbol{\Pi}},\\ \widehat{\boldsymbol{H}}_{\boldsymbol{\Pi}} &= \frac{1}{n^2}(\boldsymbol{K}\boldsymbol{K}^\top) \circ (\boldsymbol{\Pi}^\top\boldsymbol{L}\boldsymbol{L}^\top\boldsymbol{\Pi}),\\ \widehat{\boldsymbol{h}}_{\boldsymbol{\Pi}} &= \frac{1}{n}\left(\boldsymbol{K}\circ(\boldsymbol{\Pi}^\top\boldsymbol{L}\boldsymbol{\Pi})\right)\boldsymbol{1}_n.\end{aligned}$$

Consequently, LSOM is formulated as follows:

$$\max_{\boldsymbol{\Pi}} \ \mathrm{LSMI}(Z(\boldsymbol{\Pi})).$$

Since this optimization problem is in general NP-hard and is not convex, we simply use the same optimization strategy as KS-HSIC, i.e., for the current $\boldsymbol{\Pi}^{\mathrm{old}}$, the solution is updated as

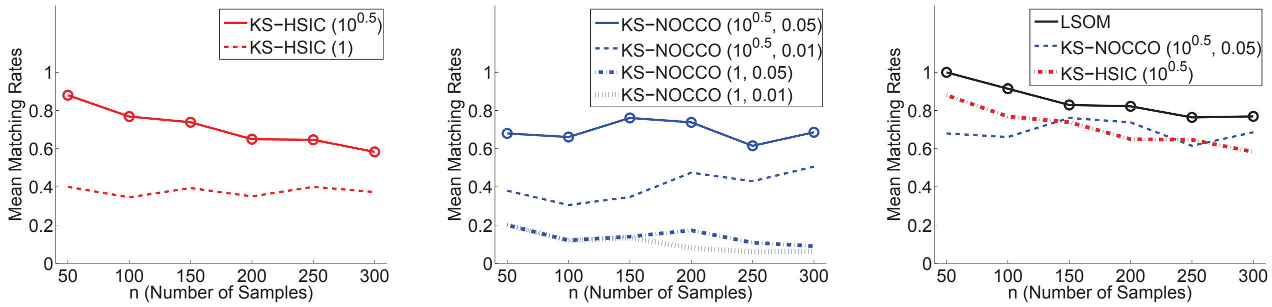
$$\begin{aligned}&\boldsymbol{\Pi}^{\mathrm{new}} =\\ &\quad (1-\eta)\boldsymbol{\Pi}^{\mathrm{old}} + \eta \operatorname*{argmax}_{\boldsymbol{\Pi}} \ \mathrm{tr}\left(\boldsymbol{\Pi}^\top\boldsymbol{L}\boldsymbol{\Pi}^{\mathrm{old}}\widehat{\boldsymbol{A}}_{\boldsymbol{\Pi}^{\mathrm{old}}}\boldsymbol{K}\right).\end{aligned} \tag{10}$$

5 Experiments

In this section, we experimentally evaluate our proposed algorithms in the image matching, unpaired voice conversion, and photo album summarization tasks.

In all the methods, we use the Gaussian kernels:

$$\begin{aligned}K(\boldsymbol{x},\boldsymbol{x}') &= \exp\left(-\frac{\|\boldsymbol{x}-\boldsymbol{x}'\|^2}{2\sigma_{\mathrm{x}}^2}\right),\\ L(\boldsymbol{y},\boldsymbol{y}') &= \exp\left(-\frac{\|\boldsymbol{y}-\boldsymbol{y}'\|^2}{2\sigma_{\mathrm{y}}^2}\right),\end{aligned}$$



(a) KS-HSIC with different Gaussian kernel widths.

(b) KS-NOCCO with different Gaussian kernel widths and regularization parameters.

(c) LSOM (tuned by CV), optimally-tuned KS-NOCCO, and optimally-tuned KS-HSIC.

Figure 1: Image matching results. The best method in terms of the mean error and comparable methods according to the t-test at the significance level 1% are specified by '∘'.

and we set the maximum number of iterations for updating permutation matrices to 20 and the step size η to 1. To avoid falling into undesirable local optima, optimization is carried out 10 times with different initial permutation matrices, which are determined by the eigenvalue-based initialization heuristic with Gaussian kernel widths

$$(\sigma_{\mathrm{x}}, \sigma_{\mathrm{y}}) = c \times (m_{\mathrm{x}}, m_{\mathrm{y}}),$$

where $c = 1^{1/2}, 2^{1/2}, \ldots, 10^{1/2}$, and

$$m_{\mathrm{x}} = 2^{-1/2}\mathrm{median}(\{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|\}_{i,j=1}^{n}),$$
$$m_{\mathrm{y}} = 2^{-1/2}\mathrm{median}(\{\|\boldsymbol{y}_i - \boldsymbol{y}_j\|\}_{i,j=1}^{n}).$$

In KS-HSIC and KS-NOCCO, we use the Gaussian kernel with the following widths:

$$(\sigma_{\mathrm{x}}, \sigma_{\mathrm{y}}) = c' \times (m_{\mathrm{x}}, m_{\mathrm{y}}),$$

where $c' = 1^{1/2}, 10^{1/2}$. In KS-NOCCO, we use the following regularization parameters:

$$\epsilon = 0.01, 0.05.$$

In LSOM, we choose the model parameters of LSMI, σ_{x}, σ_{y}, and λ by 2-fold CV from

$$(\sigma_{\mathrm{x}}, \sigma_{\mathrm{y}}) = c \times (m_{\mathrm{x}}, m_{\mathrm{y}}),$$
$$\lambda = 10^{-1}, 10^{-2}, 10^{-3}.$$

5.1 Image Matching

Let us consider a toy image matching problem. In this experiment, we use images with RGB format used in Quadrianto *et al.* (2010), which were originally extracted from *Flickr*[2]. We first convert the images from RGB to Lab space and resize them to 40 × 40 pixels. Next, we convert an image into a 4800-dimensional vector ($4800 = 40 \times 40 \times 3$). Then, we vertically divide images of size 40 × 40 pixels in the middle, and make two sets of half-images $\{\boldsymbol{x}_i\}_{i=1}^{n}$ and $\{\boldsymbol{y}_i\}_{i=1}^{n}$. Given that $\{\boldsymbol{y}_i\}_{i=1}^{n}$ is randomly permuted, the goal is to recover the correct correspondence.

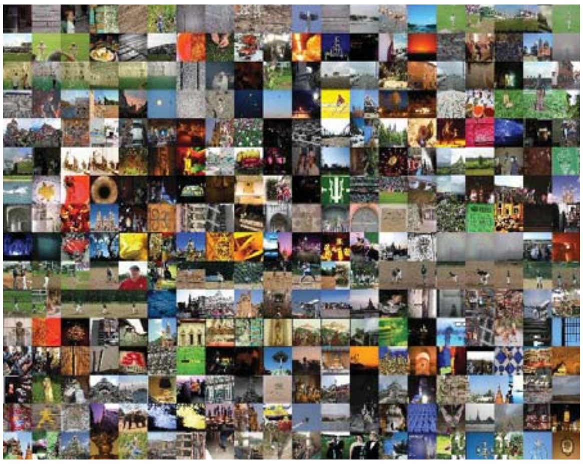


Figure 2: Image matching result by LSOM. In this case, 234 out of 320 images (73.1%) are matched correctly.

Figure 1 summarizes the average correct matching rate over 100 runs as functions of the number of images, showing that the proposed LSOM method tends to outperform the best tuned KS-NOCCO and KS-NOCCO methods. Note that the tuning parameters of LSOM (σ_{x}, σ_{y}, and λ) are automatically tuned by CV. Figure 2 depicts an example of image matching results obtained by LSOM, showing that most of the images are correctly matched.

[2] http://www.flickr.com

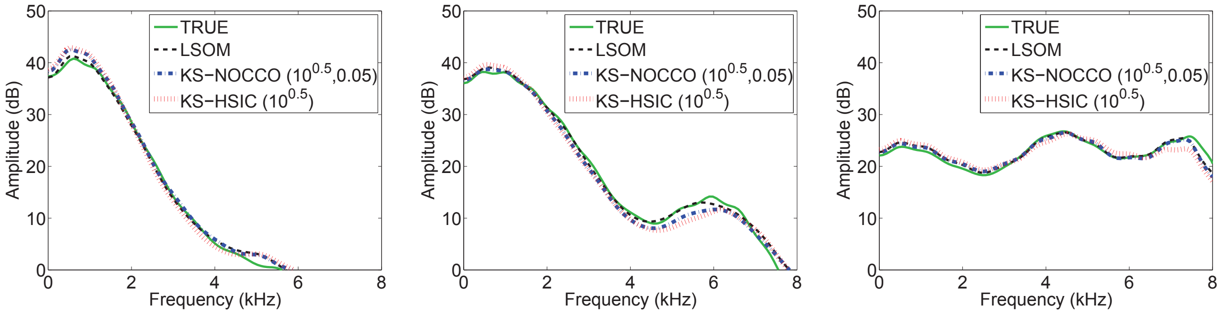


Figure 3: True spectral envelopes and their estimates.

5.2 Unpaired Voice Conversion

Next, we consider an unpaired voice conversion task, which is aimed at matching the voice of a source speaker with that of a target speaker.

In this experiment, we use 200 short utterance samples recorded from two male speakers in French, with sampling rate 44.1kHz. We first convert the utterance samples to 50-dimensional *line spectral frequencies* (LSF) vector (Kain and Macon, 1988). We denote the source and target LSF vectors by $\boldsymbol{x}$ and $\boldsymbol{y}$, respectively. Then the voice conversion task can be regarded as a multi-dimensional regression problem of learning a function from $\boldsymbol{x}$ to $\boldsymbol{y}$. However, different from a standard regression setup, paired training samples are not available; instead, only unpaired samples $\{\boldsymbol{x}_i\}_{i=1}^n$ and $\{\boldsymbol{y}_i\}_{i=1}^n$ are given.

By CDOM, we first match $\{\boldsymbol{x}_i\}_{i=1}^n$ and $\{\boldsymbol{y}_i\}_{i=1}^n$, and then we train a multi-dimensional kernel regression model (Schölkopf and Smola, 2002) using the matched samples $\{(\boldsymbol{x}_{\pi(i)}, \boldsymbol{y}_i)\}_{i=1}^n$ as

$$\min_{\boldsymbol{W}} \sum_{i=1}^{n} \|\boldsymbol{y}_i - \boldsymbol{W}^\top \boldsymbol{k}(\boldsymbol{x}_{\pi(i)})\|^2 + \frac{\delta}{2}\mathrm{tr}(\boldsymbol{W}^\top \boldsymbol{W}),$$

where

$$\boldsymbol{k}(\boldsymbol{x}) = (K(\boldsymbol{x}, \boldsymbol{x}_{\pi(1)}), \ldots, K(\boldsymbol{x}, \boldsymbol{x}_{\pi(n)}))^\top,$$
$$K(\boldsymbol{x}, \boldsymbol{x}') = \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{x}'\|^2}{2\tau^2}\right).$$

Here, τ is a Gaussian kernel width and δ is a regularization parameter; they are chosen by 2-fold CV.

We repeat the experiments 100 times by randomly shuffling training and test samples, and evaluate the voice convergence performance by *log-spectral distance* for 8000 test samples[1] (Quackenbush *et al.*, 1988). Figure 3 shows the true spectral envelope and their estimates, and Figure 4 shows the average performance over 100 runs as the number of training samples. These results show that the proposed LSOM tends to outperform KS-NOCCO and KS-HSIC.

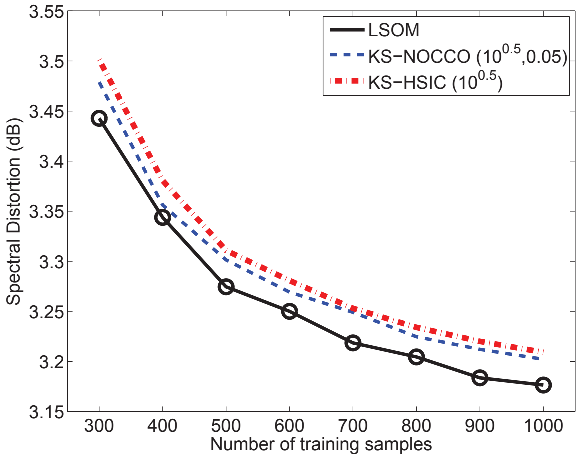


Figure 4: Unpaired voice conversion results. The best method in terms of the mean spectral distortion and comparable methods according to the t-test at the significance level 1% are specified by '○'.

5.3 Photo Album Summarization

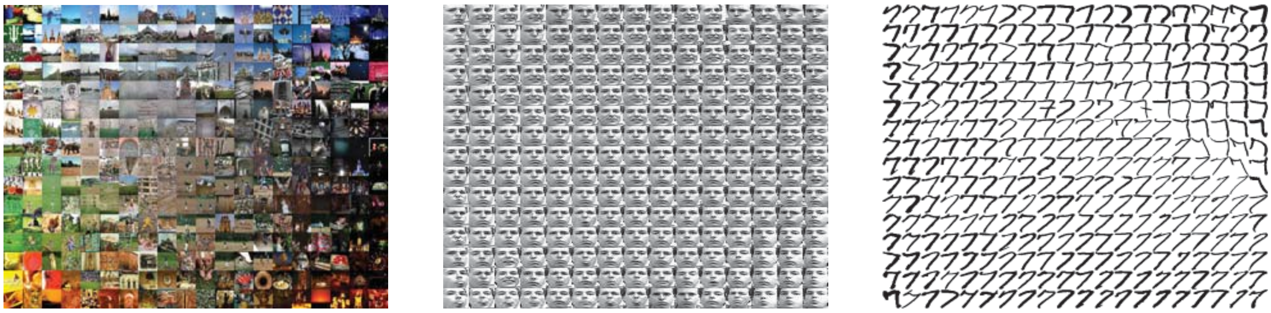
Finally, we apply the proposed LSOM method to a photo album summarization problem, where photos are automatically aligned into a designed frame expressed in the Cartesian coordinate system.

First, we use 320 images in the RGB format obtained from *Flickr*[2]. We consider a rectangular frame of 16 × 20 (= 320), and arrange the images in this rectangular frame. Figure 5(a) depicts the photo album summarization result, showing that images are aligned in the way that images with similar colors are aligned closely.

Similarly, we use the *Frey face dataset* (Roweis and

[1] The smaller the spectral distortion is, the better the quality of voice conversion is.

[2] http://www.flickr.com

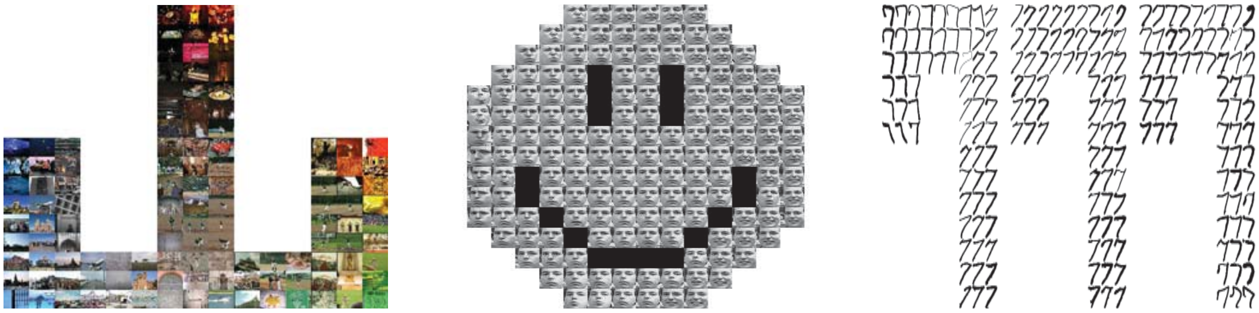


(a) Layout of 320 images into a 2D grid of size 16 by 20 using LSOM.

(b) Layout of 225 facial images into a 2D grid of size 15 by 15 using LSOM.

(c) Layout of 320 digit ‘7’ into a 2D grid of size 16 by 20 using LSOM.

Figure 5: Images are automatically aligned into rectangular grid frames expressed in the Cartesian coordinate system.



(a) Layout of 120 images into a Japanese character ‘mountain’ by LSOM.

(b) Layout of 153 facial images into ‘smiley’ by LSOM.

(c) Layout of 199 digit ‘7’ into ‘777’ by LSOM.

Figure 6: Images are automatically aligned into complex grid frames expressed in the Cartesian coordinate system.

Saul, 2000), which consists of 225 gray-scale face images with 28×20 (= 560) pixels. We similarly convert a image into a 560-dimensional vector, and we set the grid size to 15×15 (= 225). The results depicted in Figure 5(b) show that similar face images (in terms of the angle and facial expressions) are assigned in nearby cells in the grid.

Next, we apply LSOM to the USPS hand-written digit dataset (Hastie *et al.*, 2001). In this experiment, we use 320 gray-scale images of digit ‘7’ with 16×16 (= 256) pixels. We convert an image into a 256-dimensional vector, and we set the grid size to 16×20 (= 320). The result depicted in Figure 5(c) shows that digits with similar profiles are aligned closely.

Finally, we align the Flickr, Frey face, and USPS images into more complex frames—a Japanese character ‘mountain’, a smiley-face shape, and a ‘777’ digit shape. The results depicted in Figure 6 show that images with similar profiles are located in nearby grid-coordinate cells.

6 Conclusion

In this paper, we proposed two methods of cross-domain object matching (CDOM). The first method uses the dependence measure based on the normalized cross-covariance operator (NOCCO), which is advantageous over HSIC in that NOCCO is asymptotically independent of the choice of kernels. However, with finite samples, it still depends on kernels which need to be manually tuned. To cope with this problem, we proposed a more practical CDOM approach called *least-squares object matching* (LSOM). LSOM adopts *squared-loss mutual information* as a dependence measure, and it is estimated by the method of *least-squares mutual information* (LSMI). A notable advantage of the LSOM method is that it is equipped with a natural cross-validation procedure that allows us to objectively optimize tuning parameters such as the Gaussian kernel width and the regularization parameter in a data-dependent fashion. We applied the proposed methods to the image matching, unpaired voice conversion, and photo album summarization tasks, and experimentally showed that LSOM is the most promising.

Acknowledgments

We thank Dr. Fernando Villavicencio and Dr. Akisato Kimura for their valuable comments. MY acknowledges the JST PRESTO program, and MS acknowledges SCAT, AOARD, and the JST PRESTO program for financial support.

Appendix

SMI cannot be directly computed since it contains unknown densities $p(\boldsymbol{x}, \boldsymbol{y})$, $p(\boldsymbol{x})$, and $p(\boldsymbol{y})$. Here, we briefly review an SMI estimator called *least-squares mutual information* (LSMI) (Suzuki *et al.*, 2009).

Suppose that we are given n independent and identically distributed (i.i.d.) paired samples $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n$ drawn from a joint distribution with density $p(\boldsymbol{x}, \boldsymbol{y})$. A key idea of LSMI is to directly estimate the *density ratio*:

$$w(\boldsymbol{x}, \boldsymbol{y}) = \frac{p(\boldsymbol{x}, \boldsymbol{y})}{p(\boldsymbol{x})p(\boldsymbol{y})},$$

without going through density estimation of $p(\boldsymbol{x}, \boldsymbol{y})$, $p(\boldsymbol{x})$, and $p(\boldsymbol{y})$.

In LSMI, the density ratio function $w(\boldsymbol{x}, \boldsymbol{y})$ is directly modeled by the following linear model:

$$w_{\boldsymbol{\alpha}}(\boldsymbol{x}, \boldsymbol{y}) = \sum_{\ell=1}^{b} \alpha_\ell \varphi_\ell(\boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{\alpha}^\top \boldsymbol{\varphi}(\boldsymbol{x}, \boldsymbol{y}), \qquad (11)$$

where b is the number of basis functions, $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_b)^\top$ are parameters, and $\boldsymbol{\varphi}(\boldsymbol{x}, \boldsymbol{y}) = (\varphi_1(\boldsymbol{x}, \boldsymbol{y}), \ldots, \varphi_b(\boldsymbol{x}, \boldsymbol{y}))^\top$ are basis functions. Note that, we set $b = n$ in this paper.

The parameter $\boldsymbol{\alpha}$ in the model $w_{\boldsymbol{\alpha}}(\boldsymbol{x}, \boldsymbol{y})$ is learned so that the squared error between $w(\boldsymbol{x}, \boldsymbol{y})$ and $w_{\boldsymbol{\alpha}}(\boldsymbol{x}, \boldsymbol{y})$ — this is formulated as

$$\widehat{\boldsymbol{\alpha}} = \underset{\boldsymbol{\alpha}}{\mathrm{argmin}} \left[\frac{1}{2} \boldsymbol{\alpha}^\top \widehat{\boldsymbol{H}} \boldsymbol{\alpha} - \widehat{\boldsymbol{h}}^\top \boldsymbol{\alpha} + \lambda \boldsymbol{\alpha}^\top \boldsymbol{\alpha} \right],$$

where a regularization term $\lambda \boldsymbol{\alpha}^\top \boldsymbol{\alpha}$ is included for avoiding overfitting, and

$$\widehat{\boldsymbol{H}} = \frac{1}{n^2} \sum_{i,j=1}^{n} \boldsymbol{\varphi}(\boldsymbol{x}_i, \boldsymbol{y}_j) \boldsymbol{\varphi}(\boldsymbol{x}_i, \boldsymbol{y}_j)^\top,$$
$$\widehat{\boldsymbol{h}} = \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{\varphi}(\boldsymbol{x}_i, \boldsymbol{y}_i).$$

Here, we use the *product kernel* of the following form as basis functions:

$$\varphi_\ell(\boldsymbol{x}, \boldsymbol{y}) = K(\boldsymbol{x}, \boldsymbol{x}_\ell) L(\boldsymbol{y}, \boldsymbol{y}_\ell),$$

where $K(\boldsymbol{x}, \boldsymbol{x}')$ and $L(\boldsymbol{y}, \boldsymbol{y}')$ are reproducing kernels for $\boldsymbol{x}$ and $\boldsymbol{y}$.

Then $\widehat{\boldsymbol{H}}$ and $\widehat{\boldsymbol{h}}$ can be rewritten as (Petersen and Pedersen, 2008)

$$\widehat{\boldsymbol{H}} = \frac{1}{n^2} (\boldsymbol{K}\boldsymbol{K}^\top) \circ (\boldsymbol{L}\boldsymbol{L}^\top),$$
$$\widehat{\boldsymbol{h}} = \frac{1}{n} (\boldsymbol{K} \circ \boldsymbol{L}) \boldsymbol{1}_n.$$

Differentiating the above objective function with respect to $\boldsymbol{\alpha}$ and equating it to zero, we can obtain an analytic-form solution:

$$\widehat{\boldsymbol{\alpha}} = (\widehat{\boldsymbol{H}} + \lambda \boldsymbol{I}_b)^{-1} \widehat{\boldsymbol{h}}.$$

Given a density ratio estimator $\widehat{w} = w_{\widehat{\boldsymbol{\alpha}}}$, SMI can be simply approximated as

$$\mathrm{LSMI}(Z) = \frac{1}{2} \widehat{\boldsymbol{\alpha}}^\top \widehat{\boldsymbol{h}} - \frac{1}{2}.$$

In order to determine the kernel parameter and the regularization parameter λ, cross-validation (CV) is available for the LSMI estimator: First, the samples $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n$ are divided into K disjoint subsets $\{\mathcal{S}_k\}_{k=1}^K$, $\mathcal{S}_k = \{(\boldsymbol{x}_{k,i}, \boldsymbol{y}_{k,i})\}_{i=1}^{n_k}$ of (approximately) the same size, where n_k is the number of samples in the subset $\mathcal{S}_k$. Then, an estimator $\widehat{\boldsymbol{\alpha}}_{\mathcal{S}_k}$ is obtained using $\{\mathcal{S}_j\}_{j \neq k}$, and the approximation error for the hold-out samples $\mathcal{S}_k$ is computed as

$$J_{\mathcal{S}_k}^{(K\text{-CV})} = \frac{1}{2} \widehat{\boldsymbol{\alpha}}_{\mathcal{S}_k}^\top \widehat{\boldsymbol{H}}_{\mathcal{S}_k} \widehat{\boldsymbol{\alpha}}_{\mathcal{S}_k} - \widehat{\boldsymbol{h}}_{\mathcal{S}_k}^\top \widehat{\boldsymbol{\alpha}}_{\mathcal{S}_k},$$

where, for $[\boldsymbol{K}_{\mathcal{S}_k}]_{ij} = K(\boldsymbol{x}_i, \boldsymbol{x}_{k,j})$, $[\boldsymbol{L}_{\mathcal{S}_k}]_{ij} = L(\boldsymbol{y}_i, \boldsymbol{y}_{k,j})$ $i = 1, \ldots, n, j = 1, \ldots, |\mathcal{S}_k|$,

$$\widehat{\boldsymbol{H}}_{\mathcal{S}_k} = \frac{1}{n_k^2} (\boldsymbol{K}_{\mathcal{S}_k} \boldsymbol{K}_{\mathcal{S}_k}^\top) \circ (\boldsymbol{L}_{\mathcal{S}_k} \boldsymbol{L}_{\mathcal{S}_k}^\top),$$
$$\widehat{\boldsymbol{h}}_{\mathcal{S}_k} = \frac{1}{n_k} (\boldsymbol{K}_{\mathcal{S}_k} \circ \boldsymbol{L}_{\mathcal{S}_k}) \boldsymbol{1}_{n_k}.$$

This procedure is repeated for $k = 1, \ldots, K$, and its average $J^{(K\text{-CV})}$ is outputted as

$$J^{(K\text{-CV})} = \frac{1}{K} \sum_{k=1}^{K} J_{\mathcal{S}_k}^{(K\text{-CV})}.$$

We compute $J^{(K\text{-CV})}$ for all model candidates, and choose the model that minimizes $J^{(K\text{-CV})}$.

References

Cover, T. M. and Thomas, J. A. (2006). *Elements of Information Theory*. John Wiley & Sons, Inc., Hoboken, NJ, USA, 2nd edition.

Finke, G., Burkard, R. E., and Rendl, F. (1987). Quadratic assignment problems. *Annals of Discrete Mathematics*, **31**, 61–82.

Fukumizu, K., Bach, F. R., and Jordan, M. (2009a). Kernel dimension reduction in regression. *The Annals of Statistics*, **37**(4), 1871–1905.

Fukumizu, K., Gretton, A., Sun, X., and Schölkopf, B. (2009b). Kernel measures of conditional dependence. In D. Koller, D. Schuurmans, Y. Bengio, and L. Botton, editors, *Advances in Neural Information Processing Systems 21 (NIPS2008)*, pages 489–496, Cambridge, MA. MIT Press.

Gretton, A., Bousquet, O., Smola, A., and Schölkopf, B. (2005). Measuring statistical dependence with Hilbert-Schmidt norms. In *16th International Conference on Algorithmic Learning Theory (ALT 2005)*, pages 63–78.

Hardy, G. H., Littlewood, J. E., and Pólya, G. (1952). *Inequalities*. Cambridge University Press, Cambridge.

Hastie, T., Tibshirani, R., and Friedman, J. (2001). *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, New York.

Jagarlamudi, J., Juarez, S., and Daumé III, H. (2010). Kernelized sorting for natural language processing. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence (AAAI2010)*, pages 1020–1025, Atlanta, Georgia, U.S.A.

Jebara, T. (2004). Kernelized sorting, permutation, and alignment for minimum volume PCA. In *Conference on Computational Learning theory (COLT)*, pages 609–623.

Kain, A. and Macon, M. W. (1988). Spectral voice conversion for text-to-speech synthesis. In *Proceedings of 1998 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP1998)*, pages 285–288, Washington, DC, U.S.A.

Kuhn, H. (1955). The Hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, **2**(1-2), 83–97.

Kullback, S. and Leibler, R. A. (1951). On information and sufficiency. *Annals of Mathematical Statistics*, **22**, 79–86.

Minka, T. P. (2000). Old and new matrix algebra useful for statistics. Technical report, MIT Media Lab.

Pearson, K. (1900). On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling. *Philosophical Magazine*, **50**, 157–175.

Petersen, K. B. and Pedersen, M. S. (2008). The matrix cookbook. Version 20081110.

Quackenbush, S. R., Barnwell, T. P., and Clements, M. A. (1988). *Objective Measures of Speech Quality*. Prentice-Hall, Inc., Englewood Cliffs, New Jersey.

Quadrianto, N., Smola, A., Song, L., and Tuytelaars, T. (2010). Kernelized sorting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **32**, 1809–1821.

Roweis, S. T. and Saul, L. K. (2000). Nonlinear dimensionality reduction by locally linear embedding. *Science*, **290**, 2323–2326.

Schölkopf, B. and Smola, A. J. (2002). *Learning with Kernels*. MIT Press, Cambridge, MA.

Suzuki, T., Sugiyama, M., Kanamori, T., and Sese, J. (2009). Mutual information estimation reveals global associations between stimuli and biological processes. *BMC Bioinformatics*, **10**(S52).

On Information-Maximization Clustering: Tuning Parameter Selection and Analytic Solution

Masashi Sugiyama SUGI@CS.TITECH.AC.JP
Makoto Yamada YAMADA@SG.CS.TITECH.AC.JP
Manabu Kimura KIMURA@SG.CS.TITECH.AC.JP
Hirotaka Hachiya HACHIYA@SG.CS.TITECH.AC.JP
Department of Computer Science, Tokyo Institute of Technology, Tokyo, Japan.

Abstract

Information-maximization clustering learns a probabilistic classifier in an unsupervised manner so that mutual information between feature vectors and cluster assignments is maximized. A notable advantage of this approach is that it only involves continuous optimization of model parameters, which is substantially easier to solve than discrete optimization of cluster assignments. However, existing methods still involve non-convex optimization problems, and therefore finding a good local optimal solution is not straightforward in practice. In this paper, we propose an alternative information-maximization clustering method based on a *squared-loss* variant of mutual information. This novel approach gives a clustering solution *analytically* in a computationally efficient way via kernel eigenvalue decomposition. Furthermore, we provide a practical model selection procedure that allows us to objectively optimize tuning parameters included in the kernel function. Through experiments, we demonstrate the usefulness of the proposed approach.

1. Introduction

The goal of *clustering* is to classify data samples into disjoint groups in an unsupervised manner. *K-means* is a classic but still popular clustering algorithm. However, since k-means only produces linearly separated clusters, its usefulness is rather limited in practice.

To cope with this problem, various non-linear clustering methods have been developed. *Kernel k-means* (Girolami, 2002) performs k-means in a feature space induced by a reproducing kernel function. *Spectral clustering* (Shi & Malik, 2000) first unfolds non-linear data manifolds by a spectral embedding method, and then performs k-means in the embedded space. *Blurring mean-shift* (Fukunaga & Hostetler, 1975) uses a non-parametric kernel density estimator for modeling the data-generating probability density and finds clusters based on the modes of the estimated density. *Discriminative clustering* (Xu et al., 2005; Bach & Harchaoui, 2008) learns a discriminative classifier for separating clusters, where class labels are also treated as parameters to be optimized. *Dependence-maximization clustering* (Song et al., 2007; Faivishevsky & Goldberger, 2010) determines cluster assignments so that their dependence on input data is maximized.

These non-linear clustering techniques would be capable of handling highly complex real-world data. However, they suffer from lack of objective model selection strategies[1]. More specifically, the above non-linear clustering methods contain tuning parameters such as the width of Gaussian functions and the number of nearest neighbors in kernel functions or similarity measures, and these tuning parameter values need to be heuristically determined in an unsupervised manner. The problem of learning similarities/kernels was addressed in earlier works, but they considered supervised setups, i.e., labeled samples are assumed to be given. Zelnik-Manor & Perona (2005) provided a useful unsupervised heuristic to determine the similarity in a data-dependent way. However, it still requires the number of nearest neighbors to be determined man-

Appearing in *Proceedings of the 28th International Conference on Machine Learning*, Bellevue, WA, USA, 2011.

[1] 'Model selection' in this paper refers to the choice of tuning parameters in kernel functions or similarity measures, not the choice of the number of clusters.

ually (although the magic number '7' was shown to work well in their experiments).

Another line of clustering framework called *information-maximization clustering* (Agakov & Barber, 2006; Gomes et al., 2010) exhibited the state-of-the-art performance. In this information-maximization approach, probabilistic classifiers such as a kernelized Gaussian classifier (Agakov & Barber, 2006) and a kernel logistic regression classifier (Gomes et al., 2010) are learned so that *mutual information* (MI) between feature vectors and cluster assignments is maximized in an unsupervised manner. A notable advantage of this approach is that classifier training is formulated as continuous optimization problems, which are substantially simpler than discrete optimization of cluster assignments. Indeed, classifier training can be carried out in computationally efficient manners by a gradient method (Agakov & Barber, 2006) or a quasi-Newton method (Gomes et al., 2010). Furthermore, Agakov & Barber (2006) provided a model selection strategy based on the common information-maximization principle. Thus, kernel parameters can be systematically optimized in an unsupervised way.

However, in the above MI-based clustering approach, the optimization problems are non-convex, and finding a good local optimal solution is not straightforward in practice. The goal of this paper is to overcome this problem by providing a novel information-maximization clustering method. More specifically, we propose to employ a variant of MI called *squared-loss MI* (SMI), and develop a new clustering algorithm whose solution can be computed analytically in a computationally efficient way via eigenvalue decomposition. Furthermore, for kernel parameter optimization, we propose to use a non-parametric SMI estimator called *least-squares MI* (LSMI) (Suzuki et al., 2009), which was proved to achieve the optimal convergence rate with analytic-form solutions. Through experiments on various real-world datasets such as images, natural languages, accelerometric sensors, and speech, we demonstrate the usefulness of the proposed clustering method.

2. Information-Maximization Clustering with Squared-Loss Mutual Information

In this section, we describe our novel clustering algorithm.

2.1. Formulation of Information-Maximization Clustering

Suppose we are given d-dimensional i.i.d. feature vectors of size n,

$$\{\boldsymbol{x}_i \mid \boldsymbol{x}_i \in \mathbb{R}^d\}_{i=1}^n,$$

which are assumed to be drawn independently from a distribution with density $p^*(\boldsymbol{x})$. The goal of clustering is to give cluster assignments,

$$\{y_i \mid y_i \in \{1, \ldots, c\}\}_{i=1}^n,$$

to the feature vectors $\{\boldsymbol{x}_i\}_{i=1}^n$, where c denotes the number of classes. Throughout this paper, we assume that c is known.

In order to solve the clustering problem, we take the *information-maximization* approach (Agakov & Barber, 2006; Gomes et al., 2010). That is, we regard clustering as an unsupervised classification problem, and learn the class-posterior probability $p^*(y|\boldsymbol{x})$ so that 'information' between feature vector $\boldsymbol{x}$ and class label y is maximized.

The *dependence-maximization* approach (Song et al., 2007; Faivishevsky & Goldberger, 2010) is related to, but substantially different from the above information-maximization approach. In the dependence-maximization approach, cluster assignments $\{y_i\}_{i=1}^n$ are directly determined so that their dependence on feature vectors $\{\boldsymbol{x}_i\}_{i=1}^n$ is maximized. Thus, the dependence-maximization approach intrinsically involves combinatorial optimization with respect to $\{y_i\}_{i=1}^n$. On the other hand, the information-maximization approach involves continuous optimization with respect to the parameter $\boldsymbol{\alpha}$ included in a class-posterior model $p(y|\boldsymbol{x};\boldsymbol{\alpha})$. This continuous optimization of $\boldsymbol{\alpha}$ is substantially easier to solve than discrete optimization of $\{y_i\}_{i=1}^n$.

Another advantage of the information-maximization approach is that it naturally allows out-of-sample clustering based on the discriminative model $p(y|\boldsymbol{x};\boldsymbol{\alpha})$, i.e., a cluster assignment for a new feature vector can be obtained based on the learned discriminative model.

2.2. Squared-Loss Mutual Information

As an information measure, we adopt *squared-loss mutual information* (SMI). SMI between feature vector $\boldsymbol{x}$ and class label y is defined by

$$\mathrm{SMI} := \frac{1}{2} \int \sum_{y=1}^{c} p^*(\boldsymbol{x})p^*(y) \left(\frac{p^*(\boldsymbol{x}, y)}{p^*(\boldsymbol{x})p^*(y)} - 1 \right)^2 \mathrm{d}\boldsymbol{x}, \tag{1}$$

where $p^*(\boldsymbol{x}, y)$ denotes the joint density of $\boldsymbol{x}$ and y, and $p^*(y)$ is the marginal probability of y. SMI is the *Pearson divergence* (Pearson, 1900) from $p^*(\boldsymbol{x}, y)$ to $p^*(\boldsymbol{x})p^*(y)$, while the ordinary MI (Cover & Thomas, 2006) is the *Kullback-Leibler divergence* (Kullback & Leibler, 1951) from $p^*(\boldsymbol{x}, y)$ to $p^*(\boldsymbol{x})p^*(y)$:

$$\mathrm{MI} := \int \sum_{y=1}^{c} p^*(\boldsymbol{x}, y) \log \frac{p^*(\boldsymbol{x}, y)}{p^*(\boldsymbol{x})p^*(y)} \mathrm{d}\boldsymbol{x}. \tag{2}$$

The Pearson divergence and the Kullback-Leibler divergence both belong to the class of *Ali-Silvey-Csiszár divergences* (which is also known as f-divergences, see (Ali & Silvey, 1966; Csiszár, 1967)), and thus they share similar properties. For example, SMI is non-negative and takes zero if and only if $\boldsymbol{x}$ and y are statistically independent, as the ordinary MI.

In the existing information-maximization clustering methods (Agakov & Barber, 2006; Gomes et al., 2010), MI is used as the information measure. On the other hand, in this paper, we adopt SMI because it allows us to develop a clustering algorithm whose solution can be computed analytically in a computationally efficient way via eigenvalue decomposition, as described below.

2.3. Clustering by SMI Maximization

Here, we give a computationally-efficient clustering algorithm based on SMI (1).

We can express SMI as

$$\mathrm{SMI} = \frac{1}{2} \int \sum_{y=1}^{c} p^*(\boldsymbol{x}, y) \frac{p^*(\boldsymbol{x}, y)}{p^*(\boldsymbol{x})p^*(y)} \mathrm{d}\boldsymbol{x} - \frac{1}{2} \tag{3}$$

$$= \frac{1}{2} \int \sum_{y=1}^{c} p^*(y|\boldsymbol{x})p^*(\boldsymbol{x}) \frac{p^*(y|\boldsymbol{x})}{p^*(y)} \mathrm{d}\boldsymbol{x} - \frac{1}{2}. \tag{4}$$

Suppose that the class-prior probability $p^*(y)$ is set to be uniform: $p^*(y) = 1/c$. Then Eq.(4) is expressed as

$$\frac{c}{2} \int \sum_{y=1}^{c} p^*(y|\boldsymbol{x})p^*(\boldsymbol{x})p^*(y|\boldsymbol{x}) \mathrm{d}\boldsymbol{x} - \frac{1}{2}. \tag{5}$$

Let us approximate the class-posterior probability $p^*(y|\boldsymbol{x})$ by the following kernel model:

$$p(y|\boldsymbol{x}; \boldsymbol{\alpha}) := \sum_{i=1}^{n} \alpha_{y,i} K(\boldsymbol{x}, \boldsymbol{x}_i), \tag{6}$$

where $K(\boldsymbol{x}, \boldsymbol{x}')$ denotes a kernel function with a kernel parameter t. In the experiments, we will use a sparse variant of the *local-scaling kernel* (Zelnik-Manor & Perona, 2005):

$$K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \begin{cases} \exp\left(-\dfrac{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2}{2\sigma_i\sigma_j}\right) \\ \quad \text{if } \boldsymbol{x}_i \in \mathcal{N}_t(\boldsymbol{x}_j) \text{ or } \boldsymbol{x}_j \in \mathcal{N}_t(\boldsymbol{x}_i), \\ 0 \qquad\qquad\qquad\qquad \text{otherwise,} \end{cases} \tag{7}$$

where $\mathcal{N}_t(\boldsymbol{x})$ denotes the set of t nearest neighbors for $\boldsymbol{x}$ (t is the kernel parameter), σ_i is a local scaling factor defined as $\sigma_i = \|\boldsymbol{x}_i - \boldsymbol{x}_i^{(t)}\|$, and $\boldsymbol{x}_i^{(t)}$ is the t-th nearest neighbor of $\boldsymbol{x}_i$.

Further approximating the expectation with respect to $p^*(\boldsymbol{x})$ included in Eq.(5) by the empirical average of samples $\{\boldsymbol{x}_i\}_{i=1}^n$, we arrive at the following SMI approximator:

$$\widehat{\mathrm{SMI}} := \frac{c}{2n} \sum_{y=1}^{c} \boldsymbol{\alpha}_y^\top \boldsymbol{K}^2 \boldsymbol{\alpha}_y - \frac{1}{2}, \tag{8}$$

where $^\top$ denotes the transpose, $\boldsymbol{\alpha}_y := (\alpha_{y,1}, \ldots, \alpha_{y,n})^\top$, and $K_{i,j} := K(\boldsymbol{x}_i, \boldsymbol{x}_j)$.

For each cluster y, we maximize $\boldsymbol{\alpha}_y^\top \boldsymbol{K}^2 \boldsymbol{\alpha}_y$ under[2] $\|\boldsymbol{\alpha}_y\| = 1$. Since this is the *Rayleigh quotient*, the maximizer is given by the normalized principal eigenvector of $\boldsymbol{K}$ (Horn & Johnson, 1985). To avoid all the solutions $\{\boldsymbol{\alpha}_y\}_{y=1}^c$ to be reduced to the same principal eigenvector, we impose their mutual orthogonality: $\boldsymbol{\alpha}_y^\top \boldsymbol{\alpha}_{y'} = 0$ for $y \neq y'$. Then the solutions are given by the normalized eigenvectors $\boldsymbol{\phi}_1, \ldots, \boldsymbol{\phi}_c$ associated with the eigenvalues $\lambda_1 \geq \cdots \geq \lambda_n \geq 0$ of $\boldsymbol{K}$. Since the sign of $\boldsymbol{\phi}_y$ is arbitrary, we set the sign as

$$\widetilde{\boldsymbol{\phi}}_y = \boldsymbol{\phi}_y \times \mathrm{sign}(\boldsymbol{\phi}_y^\top \boldsymbol{1}_n),$$

where $\mathrm{sign}(\cdot)$ denotes the sign of a scalar and $\boldsymbol{1}_n$ denotes the n-dimensional vector with all ones.

On the other hand, since

$$p^*(y) = \int p^*(y|\boldsymbol{x})p^*(\boldsymbol{x})\mathrm{d}\boldsymbol{x} \approx \frac{1}{n} \sum_{i=1}^{n} p(y|\boldsymbol{x}_i; \boldsymbol{\alpha}) = \boldsymbol{\alpha}_y^\top \boldsymbol{K} \boldsymbol{1}_n,$$

and the class-prior probability $p^*(y)$ was set to be uniform, we have the following normalization condition:

$$\boldsymbol{\alpha}_y^\top \boldsymbol{K} \boldsymbol{1}_n = 1/c.$$

Furthermore, probability estimates should be non-negative, which can be achieved by rounding up negative outputs to zero. Taking these issues into account,

[2] Note that this unit-norm constraint is not essential since the obtained solution is renormalized later.

cluster assignments $\{y_i\}_{i=1}^n$ for $\{\boldsymbol{x}_i\}_{i=1}^n$ are determined as

$$y_i = \underset{y}{\operatorname{argmax}} \frac{[\max(\mathbf{0}_n, \widetilde{\boldsymbol{\phi}}_y)]_i}{\max(\mathbf{0}_n, \widetilde{\boldsymbol{\phi}}_y)^\top \mathbf{1}_n},$$

where the max operation for vectors is applied in the element-wise manner and $[\cdot]_i$ denotes the i-th element of a vector. Note that we used $\boldsymbol{K}\widetilde{\boldsymbol{\phi}}_y = \lambda_y \widetilde{\boldsymbol{\phi}}_y$ in the above derivation.

We call the above method *SMI-based clustering* (SMIC).

2.4. Kernel Parameter Choice by SMI Maximization

Since the above clustering approach was developed in the framework of SMI maximization, it would be natural to determine the kernel parameters so that SMI is maximized. A direct approach is to use the above SMI estimator $\widehat{\mathrm{SMI}}$ also for kernel parameter choice. However, this direct approach is not favorable because $\widehat{\mathrm{SMI}}$ is an unsupervised SMI estimator (i.e., SMI is estimated only from unlabeled samples $\{\boldsymbol{x}_i\}_{i=1}^n$). In the model selection stage, however, we have already obtained labeled samples $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$, and thus supervised estimation of SMI is possible. For supervised SMI estimation, a non-parametric SMI estimator called *least-squares mutual information* (LSMI) (Suzuki et al., 2009) was shown to achieve the optimal convergence rate. For this reason, we propose to use LSMI for model selection, instead of $\widehat{\mathrm{SMI}}$ (8).

LSMI is an estimator of SMI based on paired samples $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$. The key idea of LSMI is to learn the following *density-ratio function*,

$$r^*(\boldsymbol{x}, y) := \frac{p^*(\boldsymbol{x}, y)}{p^*(\boldsymbol{x})p^*(y)}, \tag{9}$$

without going through density estimation of $p^*(\boldsymbol{x}, y)$, $p^*(\boldsymbol{x})$, and $p^*(y)$. More specifically, let us employ the following density-ratio model:

$$r(\boldsymbol{x}, y; \boldsymbol{\theta}) := \sum_{\ell: y_\ell = y} \theta_\ell L(\boldsymbol{x}, \boldsymbol{x}_\ell), \tag{10}$$

where $L(\boldsymbol{x}, \boldsymbol{x}')$ is a kernel function with kernel parameter γ. In the experiments, we will use the Gaussian kernel:

$$L(\boldsymbol{x}, \boldsymbol{x}') = \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{x}'\|^2}{2\gamma^2}\right). \tag{11}$$

The parameter $\boldsymbol{\theta}$ in the above density-ratio model is learned so that the following squared error is minimized:

$$\frac{1}{2} \int \sum_{y=1}^{c} \Big(r(\boldsymbol{x}, y; \boldsymbol{\theta}) - r^*(\boldsymbol{x}, y) \Big)^2 p^*(\boldsymbol{x}) p^*(y) \mathrm{d}\boldsymbol{x}. \tag{12}$$

Among n cluster assignments $\{y_i\}_{i=1}^n$, let n_y be the number of samples in cluster y. Let $\boldsymbol{\theta}_y$ be the parameter vector corresponding to the kernel bases $\{L(\boldsymbol{x}, \boldsymbol{x}_\ell)\}_{\ell: y_\ell = y}$, i.e., $\boldsymbol{\theta}_y$ is the n_y-dimensional subvector of $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_n)^\top$ consisting of indices $\{\ell \mid y_\ell = y\}$. Then an empirical and regularized version of the optimization problem (12) is given for each y as follows:

$$\min_{\boldsymbol{\theta}_y} \left[\frac{1}{2} \boldsymbol{\theta}_y^\top \widehat{\boldsymbol{H}}^{(y)} \boldsymbol{\theta}_y - \boldsymbol{\theta}_y^\top \widehat{\boldsymbol{h}}^{(y)} + \delta \boldsymbol{\theta}_y^\top \boldsymbol{\theta}_y \right], \tag{13}$$

where δ (≥ 0) is the regularization parameter. $\widehat{\boldsymbol{H}}^{(y)}$ is the $n_y \times n_y$ matrix and $\widehat{\boldsymbol{h}}^{(y)}$ is the n_y-dimensional vector defined as

$$\widehat{H}_{\ell,\ell'}^{(y)} := \frac{n_y}{n^2} \sum_{i=1}^{n} L(\boldsymbol{x}_i, \boldsymbol{x}_\ell^{(y)}) L(\boldsymbol{x}_i, \boldsymbol{x}_{\ell'}^{(y)}),$$

$$\widehat{h}_\ell^{(y)} := \frac{1}{n} \sum_{i: y_i = y} L(\boldsymbol{x}_i, \boldsymbol{x}_\ell^{(y)}),$$

where $\boldsymbol{x}_\ell^{(y)}$ is the ℓ-th sample in class y (which corresponds to $\widehat{\theta}_\ell^{(y)}$).

A notable advantage of LSMI is that the solution $\widehat{\boldsymbol{\theta}}^{(y)}$ can be computed analytically as

$$\widehat{\boldsymbol{\theta}}^{(y)} = (\widehat{\boldsymbol{H}}^{(y)} + \delta \boldsymbol{I})^{-1} \widehat{\boldsymbol{h}}^{(y)}.$$

Then a density-ratio estimator is obtained analytically as follows:

$$\widehat{r}(\boldsymbol{x}, y) = \sum_{\ell=1}^{n_y} \widehat{\theta}_\ell^{(y)} L(\boldsymbol{x}, \boldsymbol{x}_\ell^{(y)}).$$

The accuracy of the above least-squares density-ratio estimator depends on the choice of the kernel parameter γ and the regularization parameter δ. They can be systematically optimized based on cross-validation as follows (Suzuki et al., 2009). The samples $\mathcal{Z} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$ are divided into M disjoint subsets $\{\mathcal{Z}_m\}_{m=1}^M$ of approximately the same size. Then a density-ratio estimator $\widehat{r}_m(\boldsymbol{x}, y)$ is obtained using $\mathcal{Z} \backslash \mathcal{Z}_m$ (i.e., all samples without $\mathcal{Z}_m$), and its out-of-sample error (which corresponds to Eq.(12) without irrelevant constant) for the hold-out samples $\mathcal{Z}_m$ is computed as

$$\mathrm{CV}_m := \frac{1}{2|\mathcal{Z}_m|^2} \sum_{\boldsymbol{x}, y \in \mathcal{Z}_m} \widehat{r}_m(\boldsymbol{x}, y)^2 - \frac{1}{|\mathcal{Z}_m|} \sum_{(\boldsymbol{x}, y) \in \mathcal{Z}_m} \widehat{r}_m(\boldsymbol{x}, y).$$

This procedure is repeated for $m = 1, \dots, M$, and the average of the above hold-out error over all m is computed. Finally, the kernel parameter γ and the regularization parameter δ that minimize the average hold-out error are chosen as the most suitable ones.

Based on the expression of SMI given by Eq.(3), an SMI estimator called LSMI is given as follows:

$$\mathrm{LSMI} := \frac{1}{2n}\sum_{i=1}^{n} \widehat{r}(\boldsymbol{x}_i, y_i) - \frac{1}{2}, \qquad (14)$$

where $\widehat{r}(\boldsymbol{x}, y)$ is a density-ratio estimator obtained above. Since $\widehat{r}(\boldsymbol{x}, y)$ can be computed analytically, LSMI can also be computed analytically.

We use LSMI for model selection of SMIC. More specifically, we compute LSMI as a function of the kernel parameter t of $K(\boldsymbol{x}, \boldsymbol{x}')$ included in the cluster-posterior model (6), and choose the one that maximizes LSMI.

MATLAB implementation of the proposed clustering method is available from 'http://sugiyama-www.cs.titech.ac.jp/~sugi/software/SMIC'.

3. Existing Methods

In this section, we qualitatively compare the proposed approach with existing methods.

3.1. Spectral Clustering

The basic idea of *spectral clustering* (Shi & Malik, 2000) is to first unfold non-linear data manifolds by a spectral embedding method, and then perform k-means in the embedded space. More specifically, given sample-sample similarity $W_{i,j} \geq 0$, the minimizer of the following criterion with respect to $\{\boldsymbol{\xi}_i\}_{i=1}^n$ is obtained under some normalization constraint:

$$\sum_{i,j}^{n} W_{i,j} \left\| \frac{1}{\sqrt{D_{i,i}}}\boldsymbol{\xi}_i - \frac{1}{\sqrt{D_{j,j}}}\boldsymbol{\xi}_j \right\|^2,$$

where $\boldsymbol{D}$ is the diagonal matrix with i-th diagonal element given by $D_{i,i} := \sum_{j=1}^n W_{i,j}$. Consequently, the embedded samples are given by the principal eigenvectors of $\boldsymbol{D}^{-\frac{1}{2}}\boldsymbol{W}\boldsymbol{D}^{-\frac{1}{2}}$, followed by normalization. Note that spectral clustering was shown to be equivalent to a weighted variant of kernel k-means with some specific kernel (Dhillon et al., 2004).

The performance of spectral clustering depends heavily on the choice of sample-sample similarity $W_{i,j}$. Zelnik-Manor & Perona (2005) proposed a useful unsupervised heuristic to determine the similarity in a data-dependent manner, called *local scaling*: $W_{i,j} = \exp\left(-\frac{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2}{2\sigma_i\sigma_j}\right)$, where σ_i is a local scaling factor defined as $\sigma_i = \|\boldsymbol{x}_i - \boldsymbol{x}_i^{(t)}\|$, and $\boldsymbol{x}_i^{(t)}$ is the t-th nearest neighbor of $\boldsymbol{x}_i$. t is the tuning parameter in the local scaling similarity, and $t = 7$ was shown to be useful (Zelnik-Manor & Perona, 2005; Sugiyama, 2007). However, this magic number '7' does not seem to work always well in general.

If $\boldsymbol{D}^{-\frac{1}{2}}\boldsymbol{W}\boldsymbol{D}^{-\frac{1}{2}}$ is regarded as a kernel matrix, spectral clustering will be similar to the proposed SMIC method described in Section 2.3. However, SMIC does not require the post k-means processing since the principal components have clear interpretation as parameter estimates of the class-posterior model (6). Furthermore, our proposed approach provides a systematic model selection strategy, which is a notable advantage over spectral clustering.

3.2. Blurring Mean-Shift Clustering

Blurring mean-shift (Fukunaga & Hostetler, 1975) is a non-parametric clustering method based on the *modes* of the data-generating probability density.

In the blurring mean-shift algorithm, a kernel density estimator (Silverman, 1986) is used for modeling the data-generating probability density:

$$\widehat{p}(\boldsymbol{x}) = \frac{1}{n}\sum_{i=1}^{n} K\left(\|\boldsymbol{x} - \boldsymbol{x}_i\|^2/\sigma^2\right),$$

where $K(\xi)$ is a kernel function such as a Gaussian kernel $K(\xi) = e^{-\xi/2}$. Taking the derivative of $\widehat{p}(\boldsymbol{x})$ with respect to $\boldsymbol{x}$ and equating the derivative at $\boldsymbol{x} = \boldsymbol{x}_i$ to zero, we obtain the following updating formula for sample $\boldsymbol{x}_i$ $(i = 1, \dots, n)$:

$$\boldsymbol{x}_i \longleftarrow \frac{\sum_{j=1}^n W_{i,j}\boldsymbol{x}_j}{\sum_{j'=1}^n W_{i,j'}},$$

where $W_{i,j} := K'\left(\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2/\sigma^2\right)$ and $K'(\xi)$ is the derivative of $K(\xi)$. Each mode of the density is regarded as a representative of a cluster, and each data point is assigned to the cluster which it converges to.

Carreira-Perpiñán (2007) showed that the blurring mean-shift algorithm can be interpreted as an *EM algorithm* (Dempster et al., 1977), where $W_{i,j}/(\sum_{j'=1}^n W_{i,j'})$ is regarded as the posterior probability of the i-th sample belonging to the j-th cluster. Furthermore, the above update rule can be expressed in a matrix form as $\boldsymbol{X} \longleftarrow \boldsymbol{X}\boldsymbol{P}$, where $\boldsymbol{X} = (\boldsymbol{x}_1, \dots, \boldsymbol{x}_n)$ is a sample matrix and $\boldsymbol{P} := \boldsymbol{W}\boldsymbol{D}^{-1}$ is a *stochastic matrix* of the random walk in a graph with adjacency $\boldsymbol{W}$ (Chung, 1997). $\boldsymbol{D}$ is defined as

$D_{i,i} := \sum_{j=1}^{n} W_{i,j}$ and $D_{i,j} = 0$ for $i \neq j$. If $\boldsymbol{P}$ is independent of $\boldsymbol{X}$, the above iterative algorithm corresponds to the *power method* (Golub & Loan, 1996) for finding the leading left eigenvector of $\boldsymbol{P}$. Then, this algorithm is highly related to the spectral clustering which computes the principal eigenvectors of $\boldsymbol{D}^{-\frac{1}{2}}\boldsymbol{W}\boldsymbol{D}^{-\frac{1}{2}}$ (see Section 3.1). Although $\boldsymbol{P}$ depends on $\boldsymbol{X}$ in reality, Carreira-Perpiñán (2006) insisted that this analysis is still valid since $\boldsymbol{P}$ and $\boldsymbol{X}$ quickly reach a quasi-stable state.

An attractive property of blurring mean-shift is that the number of clusters is automatically determined as the number of modes in the probability density estimate. However, this choice depends on the kernel parameter σ and there is no systematic way to determine σ, which is restrictive compared with the proposed method. Another critical drawback of the blurring mean-shift algorithm is that it eventually converges to a single point (Cheng, 1995), and therefore a sensible stopping criterion is necessary in practice. Although Carreira-Perpiñán (2006) gave a useful heuristic for stopping the iteration, it is not clear whether this heuristic always works well in practice.

4. Experiments

In this section, we experimentally evaluate the performance of the proposed and existing clustering methods.

4.1. Illustration

First, we illustrate the behavior of the proposed method using artificial datasets described in the top row of Figure 1. The dimensionality is $d = 2$ and the sample size is $n = 200$. As a kernel function, we used the sparse local-scaling kernel (7) for SMIC, where the kernel parameter t was chosen from $\{1, \ldots, 10\}$ based on LSMI with the Gaussian kernel (11).

The top graphs in Figure 1 depict the cluster assignments obtained by SMIC, and the bottom graphs in Figure 1 depict the model selection curves obtained by LSMI. The results show that SMIC combined with LSMI works well for these toy datasets.

4.2. Performance Comparison

Next, we systematically compare the performance of the proposed and existing clustering methods using various real-world datasets such as images, natural languages, accelerometric sensors, and speech.

We compared the performance of the following methods, which all do not contain open tuning parameters and therefore experimental results are fair and objective: K-means (KM), spectral clustering with the self-tuning local-scaling similarity (SC) (Zelnik-Manor & Perona, 2005), mean nearest-neighbor clustering (MNN) (Faivishevsky & Goldberger, 2010), MI-based clustering for kernel logistic models (MIC) (Gomes et al., 2010) with model selection by *maximum-likelihood MI* (Suzuki et al., 2008), and the proposed SMIC.

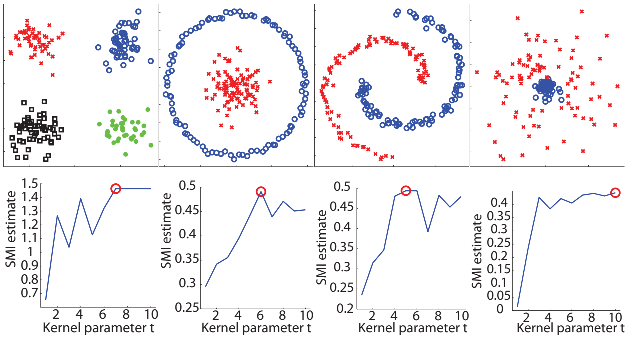


Figure 1. Illustrative examples. Cluster assignments obtained by SMIC (top) and model selection curves obtained by LSMI (bottom).

The clustering performance was evaluated by the *adjusted Rand index* (ARI) (Hubert & Arabie, 1985) between inferred cluster assignments and the ground truth categories. Larger ARI values mean better performance, and ARI takes its maximum value 1 when two sets of cluster assignments are identical. In addition, we also evaluated the computational efficiency of each method by the CPU computation time.

We used various real-world datasets including images, natural languages, accelerometric sensors, and speech: The *USPS* hand-written digit dataset ('digit'), the *Olivetti Face* dataset ('face'), the *20-Newsgroups* dataset ('document'), the *SENSEVAL-2* dataset ('word'), the *ALKAN* dataset ('accelerometry'), and the in-house speech dataset ('speech'). Detailed explanation of the datasets is omitted due to lack of space.

For each dataset, the experiment was repeated 100 times with random choice of samples from a pool. Samples were centralized and their variance was normalized in the dimension-wise manner, before feeding them to clustering algorithms.

The experimental results are described in Table 1. For the *digit* dataset, MIC and SMIC outperform KM, SC, and MNN in terms of ARI. The entire computation time of SMIC including model selection is faster than KM, SC, and MIC, and is comparable to MNN which does not include a model selection procedure. For the

Table 1. Experimental results on real-world datasets (with equal cluster size). The average clustering accuracy (and its standard deviation in the bracket) in terms of ARI and the average CPU computation time in second over 100 runs are described. The best method in terms of the average ARI and methods judged to be comparable to the best one by the *t-test* at the significance level 1% are described in boldface. Computation time of MIC and SMIC corresponds to the time for computing a clustering solution after model selection has been carried out. For references, computation time for the entire procedure including model selection is described in the square bracket.

Digit ($d = 256,\ n = 5000,\ \text{and}\ c = 10$)

	KM	SC	MNN	MIC	SMIC
ARI	0.42(0.01)	0.24(0.02)	0.44(0.03)	**0.63(0.08)**	**0.63(0.05)**
Time	835.9	973.3	318.5	84.4[3631.7]	14.4[359.5]

Face ($d = 4096,\ n = 100,\ \text{and}\ c = 10$)

	KM	SC	MNN	MIC	SMIC
ARI	0.60(0.11)	**0.62(0.11)**	0.47(0.10)	**0.64(0.12)**	**0.65(0.11)**
Time	93.3	2.1	1.0	1.4[30.8]	0.0[19.3]

Document ($d = 50,\ n = 700,\ \text{and}\ c = 7$)

	KM	SC	MNN	MIC	SMIC
ARI	0.00(0.00)	0.09(0.02)	0.09(0.02)	0.01(0.02)	**0.19(0.03)**
Time	77.8	9.7	6.4	3.4[530.5]	0.3[115.3]

Word ($d = 50,\ n = 300,\ \text{and}\ c = 3$)

	KM	SC	MNN	MIC	SMIC
ARI	0.04(0.05)	0.02(0.01)	0.02(0.02)	0.04(0.04)	**0.08(0.05)**
Time	6.5	5.9	2.2	1.0[369.6]	0.2[203.9]

Accelerometry ($d = 5,\ n = 300,\ \text{and}\ c = 3$)

	KM	SC	MNN	MIC	SMIC
ARI	0.49(0.04)	0.58(0.14)	**0.71(0.05)**	0.57(0.23)	**0.68(0.12)**
Time	0.4	3.3	1.9	0.8[410.6]	0.2[92.6]

Speech ($d = 50,\ n = 400,\ \text{and}\ c = 2$)

	KM	SC	MNN	MIC	SMIC
ARI	0.00(0.00)	0.00(0.00)	0.04(0.15)	**0.18(0.16)**	**0.21(0.25)**
Time	0.9	4.2	1.8	0.7[413.4]	0.3[179.7]

face dataset, SC, MIC, and SMIC are comparable to each other and are better than KM and MNN in terms of ARI. For the *document* and *word* datasets, SMIC tends to outperform the other methods. For the *accelerometry* dataset, MNN and SMIC work better than the other methods. Finally, for the *speech* dataset, MIC and SMIC work comparably well, and are significantly better than KM, SC, and MNN.

Overall, MIC was shown to work reasonably well, implying that model selectoin by maximum-likelihood MI is practically useful. SMIC was shown to work even better than MIC, with much less computation time. The accuracy improvement of SMIC over MIC was gained by computing the SMIC solution in a closed-form without any heuristic initialization. The computational efficiency of SMIC was brought by the analytic computation of the optimal solution and the class-wise optimization of LSMI (see Section 2.4).

The performance of MNN and SC was rather unstable because of the heuristic averaging of the number of nearest neighbors and the heuristic choice of local scaling. In terms of computation time, they are relatively efficient for small- to medium-sized datasets, but they are expensive for the largest dataset, *digit*. KM was not reliable for the *document* and *speech* datasets because of the restriction that the cluster boundaries are linear. For the *digit*, *face*, and *document* datasets, KM was computationally very expensive since a large number of iterations were needed until convergence to a local optimum solution.

Table 2. Experimental results on real-world datasets under imbalanced setup. ARI values are described in the table. Class-imbalance was realized by setting the sample size of the first class m times larger than other classes. The results for $m = 1$ are the same as the ones reported in Table 1.

Digit ($d = 256,\ n = 5000,\ \text{and}\ c = 10$)

	KM	SC	MNN	MIC	SMIC
$m = 1$	0.42(0.01)	0.24(0.02)	0.44(0.03)	**0.63(0.08)**	**0.63(0.05)**
$m = 2$	0.52(0.01)	0.21(0.02)	0.43(0.04)	0.60(0.05)	**0.63(0.05)**

Document ($d = 50,\ n = 700,\ \text{and}\ c = 7$)

	KM	SC	MNN	MIC	SMIC
$m = 1$	0.00(0.00)	0.09(0.02)	0.09(0.02)	0.01(0.02)	**0.19(0.03)**
$m = 2$	0.01(0.01)	0.10(0.03)	0.10(0.02)	0.01(0.02)	**0.19(0.04)**
$m = 3$	0.01(0.01)	0.10(0.03)	0.09(0.02)	-0.01(0.03)	**0.16(0.05)**
$m = 4$	0.02(0.01)	0.09(0.03)	0.08(0.02)	-0.00(0.04)	**0.14(0.05)**

Word ($d = 50,\ n = 300,\ \text{and}\ c = 3$)

	KM	SC	MNN	MIC	SMIC
$m = 1$	0.04(0.05)	0.02(0.01)	0.02(0.02)	0.04(0.04)	**0.08(0.05)**
$m = 2$	0.00(0.07)	-0.01(0.01)	0.01(0.02)	-0.02(0.05)	**0.03(0.05)**

Accelerometry ($d = 5,\ n = 300,\ \text{and}\ c = 3$)

	KM	SC	MNN	MIC	SMIC
$m = 1$	0.49(0.04)	0.58(0.14)	**0.71(0.05)**	0.57(0.23)	**0.68(0.12)**
$m = 2$	0.48(0.05)	0.54(0.14)	0.58(0.11)	0.49(0.19)	**0.69(0.16)**
$m = 3$	0.49(0.05)	0.47(0.10)	0.42(0.12)	0.42(0.14)	**0.66(0.20)**
$m = 4$	0.49(0.06)	0.38(0.11)	0.31(0.09)	0.40(0.18)	**0.56(0.22)**

Finally, we performed similar experiments under imbalanced setup, where the the sample size of the first class was set to be m times larger than other classes. The results are summarized in Table 2, showing that the performance of all methods tends to be degraded as the degree of imbalance increases. Thus, clustering becomes more challenging if the cluster size is imbalanced. Among the compared methods, the proposed SMIC still worked better than other methods.

Overall, the proposed SMIC combined with LSMI was shown to be a useful alternative to existing clustering approaches.

5. Conclusions

In this paper, we proposed a novel *information-maximization clustering* method, which learns class-posterior probabilities in an unsupervised manner so that the *squared-loss mutual information* (SMI) between feature vectors and cluster assignments is maximized. The proposed algorithm called *SMI-based clustering* (SMIC) allows us to obtain clustering solutions *analytically* by solving a kernel eigenvalue problem. Thus, unlike the previous information-maximization clustering methods (Agakov & Barber, 2006; Gomes et al., 2010), SMIC does not suffer from the problem of local optima. Furthermore, we proposed to use an optimal non-parametric SMI estimator called *least-squares mutual information* (LSMI) for data-driven parameter optimization. Through experiments, SMIC combined with LSMI was demonstrated to compare favorably with existing clustering methods.

Acknowledgments

We would like to thank Ryan Gomes for providing us his program code of information-maximization clustering. MS was supported by SCAT, AOARD, and the FIRST program. MY and MK were supported by the JST PRESTO program, and HH was supported by the FIRST program.

References

Agakov, F. and Barber, D. Kernelized infomax clustering. *NIPS 18*, pp. 17–24. MIT Press, 2006.

Ali, S. M. and Silvey, S. D. A general class of coefficients of divergence of one distribution from another. *Journal of the Royal Statistical Society, Series B*, 28(1):131–142, 1966.

Bach, F. and Harchaoui, Z. DIFFRAC: A discriminative and flexible framework for clustering. *NIPS 20*, pp. 49–56, 2008.

Carreira-Perpiñán, M. Á. Fast nonparametric clustering with Gaussian blurring mean-shift. *ICML*, pp. 153–160, 2006.

Carreira-Perpiñán, M. Á. Gaussian mean shift is an EM algorithm. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29:767–776, 2007.

Cheng, Y. Mean shift, mode seeking, and clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17:790–799, 1995.

Chung, F. R. K. *Spectral Graph Theory.* American Mathematical Society, Providence, 1997.

Cover, T. M. and Thomas, J. A. *Elements of Information Theory.* John Wiley & Sons, Inc., 2nd edition, 2006.

Csiszár, I. Information-type measures of difference of probability distributions and indirect observation. *Studia Scientiarum Mathematicarum Hungarica*, 2:229–318, 1967.

Dempster, A. P., Laird, N. M., and Rubin, D. B. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, series B*, 39(1): 1–38, 1977.

Dhillon, I. S., Guan, Y., and Kulis, B. Kernel k-means, spectral clustering and normalized cuts. *ACM SIGKDD*, pp. 551–556, 2004.

Faivishevsky, L. and Goldberger, J. A nonparametric information theoretic clustering algorithm. *ICML*, pp. 351–358, 2010.

Fukunaga, K. and Hostetler, L. D. The estimation of the gradient of a density function, with application in pattern recognition. *IEEE Transactions on Information Theory*, 21(1):32–40, 1975.

Girolami, M. Mercer kernel-based clustering in feature space. *IEEE Transactions on Neural Networks*, 13(3): 780–784, 2002.

Golub, G. H. and Loan, C. F. Van. *Matrix Computations.* Johns Hopkins University Press, 1996.

Gomes, R., Krause, A., and Perona, P. Discriminative clustering by regularized information maximization. *NIPS 23*, pp. 766–774. 2010.

Horn, R. A. and Johnson, C. A. *Matrix Analysis.* Cambridge University Press, 1985.

Hubert, L. and Arabie, P. Comparing partitions. *Journal of Classification*, 2(1):193–218, 1985.

Kullback, S. and Leibler, R. A. On information and sufficiency. *Annals of Mathematical Statistics*, 22:79–86, 1951.

Pearson, K. On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling. *Philosophical Magazine*, 50:157–175, 1900.

Shi, J. and Malik, J. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):888–905, 2000.

Silverman, B. W. *Density Estimation for Statistics and Data Analysis.* Chapman and Hall, 1986.

Song, L., Smola, A., Gretton, A., and Borgwardt, K. A dependence maximization view of clustering. *ICML*, pp. 815–822, 2007.

Sugiyama, M. Dimensionality reduction of multimodal labeled data by local Fisher discriminant analysis. *Journal of Machine Learning Research*, 8:1027–1061, 2007.

Suzuki, T., Sugiyama, M., Sese, J., and Kanamori, T. Approximating mutual information by maximum likelihood density ratio estimation. *JMLR Workshop and Conference Proceedings*, 4:5–20, 2008.

Suzuki, T., Sugiyama, M., Kanamori, T., and Sese, J. Mutual information estimation reveals global associations between stimuli and biological processes. *BMC Bioinformatics*, 10(1):S52, 2009.

Xu, L., Neufeld, J., Larson, B., and Schuurmans, D. Maximum margin clustering. *NIPS 17*, pp. 1537–1544. 2005.

Zelnik-Manor, L. and Perona, P. Self-tuning spectral clustering. *NIPS 17*, pp. 1601–1608, 2005.

Semi-Supervised Learning of Class Balance under Class-Prior Change by Distribution Matching

Marthinus Christoffel du Plessis CHRISTO@SG.CS.TITECH.AC.JP
Masashi Sugiyama SUGI@CS.TITECH.AC.JP
Department of Computer Science, Tokyo Institute of Technology, Tokyo, Japan

Abstract

In real-world classification problems, the class balance in the training dataset does not necessarily reflect that of the test dataset, which can cause significant estimation bias. If the class ratio of the test dataset is known, instance re-weighting or resampling allows systematical bias correction. However, learning the class ratio of the test dataset is challenging when no labeled data is available from the test domain. In this paper, we propose to estimate the class ratio in the test dataset by matching probability distributions of training and test input data. We demonstrate the utility of the proposed approach through experiments.

1. Introduction

Most supervised learning algorithms assume that training and test data follow the same probability distribution (Vapnik, 1998; Hastie et al., 2001; Bishop, 2006). However, this de facto standard assumption is often violated in real-world problems, caused by intrinsic sample selection bias or inevitable non-stationarity (Heckman, 1979; Quiñonero-Candela et al., 2009; Sugiyama & Kawanabe, 2012).

In classification scenarios, changes in class balance are often observed—for example, the male-female ratio is almost fifty-fifty in the real-world (test set), whereas training samples collected in a research laboratory tends to be dominated by male data. Such a situation is called a *class-prior change*, and the bias caused by differing class balances can be systematically adjusted by instance re-weighting or resampling if the class balance in the test dataset is known (Elkan, 2001; Lin et al., 2002).

However, the class ratio in the test dataset is often unknown in practice. A possible approach to coping with this problem is to learn a classifier so that the performance for all possible class balances is improved, e.g., through maximization of the area under the ROC curve (Cortes & Mohri, 2004; Clémençon et al., 2009). Another, possibly more direct approach is to estimate the class ratio in the test dataset and use the estimates for instance re-weighting or resampling. In this paper, we focus on the latter scenario under a semi-supervised learning setup (Chapelle et al., 2006), where no labeled data is available from the test domain.

Saerens et al. (2001) is a seminal paper on this topic, which proposed to estimate the class ratio by the expectation-maximization (EM) algorithm (Dempster et al., 1977)—alternately updating the test class-prior and class-posterior probabilities from some initial estimates until convergence. This method has been successfully applied to various real-world problems such as word sense disambiguation (Chan & Ng, 2006) and remote sensing (Latinne et al., 2001).

In this paper, we first reformulate the above algorithm, and show that this actually corresponds to approximating the test input distribution by a linear combination of class-wise input distributions under the Kullback-Leibler (KL) divergence (Kullback & Leibler, 1951). In this procedure, the class-wise input distributions are approximated via class-posterior estimation, for example, by kernel logistic regression (Hastie et al., 2001) or its squared-loss variant (Sugiyama, 2010).

This new formulation motivates us to develop a new approach, since indirectly estimating the divergence by estimating the individual class-posterior distributions may not be the best scheme. Recently, KL divergence estimation based on *direct density-ratio estimation* has been shown to be promising (Nguyen et al., 2010; Sugiyama et al., 2008). Furthermore, a squared-loss variant of the KL divergence called the Pearson (PE) divergence (Pearson, 1900) can also be approximated in the same way, with an analytic solution that can be computed efficiently (Kanamori et al.,

Appearing in *Proceedings of the 29^{th} International Conference on Machine Learning*, Edinburgh, Scotland, UK, 2012. Copyright 2012 by the author(s)/owner(s).

2009a). The PE divergence and the KL divergence both belong to the f-divergence class (Ali & Silvey, 1966; Csiszár, 1967), which share similar properties. In this paper, with the aid of this density-ratio based PE divergence estimator, we propose a new semi-supervised method for estimating the class ratio in the test dataset. Through experiments, we demonstrate the usefulness of the proposed method.

2. Problem Formulation and Existing Method

In this section, we formulate the problem of semi-supervised class-prior estimation and review an existing method (Saerens et al., 2001).

2.1. Problem Formulation

Let $\boldsymbol{x} \in \mathbb{R}^d$ be the d-dimensional input data, $y \in \{1, \ldots, c\}$ be the class label, and c be the number of classes. We consider class-prior change, i.e., the class-prior probability for training data $p(y)$ and that for test data $p'(y)$ are different. However, we assume that the class-conditional density for training data $p(\boldsymbol{x}|y)$ and that for test data $p'(\boldsymbol{x}|y)$ are the same:

$$p(\boldsymbol{x}|y) = p'(\boldsymbol{x}|y). \tag{1}$$

Note that training and test joint densities $p(\boldsymbol{x}, y)$ and $p'(\boldsymbol{x}, y)$ as well as training and test input densities $p(\boldsymbol{x})$ and $p'(\boldsymbol{x})$ are generally different under this setup.

The goal of this paper is to estimate $p'(y)$ from labeled training samples $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^{n}$ drawn independently from $p(\boldsymbol{x}, y)$ and unlabeled test samples $\{\boldsymbol{x}'_i\}_{i=1}^{n'}$ drawn independently from $p'(\boldsymbol{x})$. Given test labels $\{y'_i\}_{i=1}^{n'}$, $p'(y)$ can be naively estimated by n'_y/n', where n'_y is the number of test samples in class y. Here, however, we would like to estimate $p'(y)$ *without* $\{y'_i\}_{i=1}^{n'}$.

2.2. Existing Method

We give a brief overview of an existing method for semi-supervised class-prior estimation (Saerens et al., 2001), which is based on the expectation-maximization (EM) algorithm (Dempster et al., 1977).

In the algorithm, test class-prior and class-posterior estimates $\widehat{p}'(y)$ and $\widehat{p}'(y|\boldsymbol{x})$ are iteratively updated as follows:

1. Obtain an estimate of the training class-posterior probability, $\widehat{p}(y|\boldsymbol{x})$, from training data $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^{n}$, for example, by kernel logistic regression (Hastie et al., 2001) or its squared-loss variant (Sugiyama, 2010).

2. Obtain an estimate of the training class-prior probability, $\widehat{p}(y)$, from the labeled training data $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^{n}$ as $\widehat{p}(y) = n_y/n$, where n_y is the number of training samples in class y. Set the initial estimate of the test class-posterior probability equal to it: $\widehat{p}'_0(y) = \widehat{p}(y)$.

3. Repeat until convergence: $t = 1, 2, \ldots$
   (a) Compute a new test class-posterior estimate $\widehat{p}'_t(y|\boldsymbol{x})$ based on the current test class-prior estimate $\widehat{p}'_{t-1}(y)$ as
   $$\widehat{p}'_t(y|\boldsymbol{x}) = \frac{\widehat{p}'_{t-1}(y)\widehat{p}(y|\boldsymbol{x})/\widehat{p}(y)}{\sum_{y'=1}^{c} \widehat{p}'_{t-1}(y')\widehat{p}(y'|\boldsymbol{x})/\widehat{p}(y')}. \tag{2}$$
   (b) Compute a new test class-prior estimate $\widehat{p}'_t(y)$ based on the current test class-prior estimate $\widehat{p}'_t(y|\boldsymbol{x})$ as
   $$\widehat{p}'_t(y) = \frac{1}{n'}\sum_{i=1}^{n'} \widehat{p}'_t(y|\boldsymbol{x}'_i). \tag{3}$$

This procedure was shown to converge to a local optimal solution.

Note that Eq.(2) comes from the Bayes formulae,

$$p(\boldsymbol{x}|y) = \frac{p(y|\boldsymbol{x})p(\boldsymbol{x})}{p(y)} \text{ and } p'(\boldsymbol{x}|y) = \frac{p'(y|\boldsymbol{x})p'(\boldsymbol{x})}{p'(y)},$$

combined with Eq.(1):

$$p'(y|\boldsymbol{x}) \propto \frac{p'(y)}{p(y)} p(y|\boldsymbol{x}).$$

Eq.(3) comes from empirical marginalization of

$$p'(y) = \int p'(y|\boldsymbol{x})p'(\boldsymbol{x})\mathrm{d}\boldsymbol{x}.$$

3. Reformulation of the EM Algorithm as Distribution Matching

In this section, we show that the above EM algorithm can be interpreted as matching the test input density to a linear combination of class-wise input distributions under the Kullback-Leibler (KL) divergence (Kullback & Leibler, 1951).

Based on the assumption that the class-conditional densities for training and test data are unchanged (see Eq.(1)), let us model the test input density $p'(\boldsymbol{x})$ by

$$q'(\boldsymbol{x}) = \sum_{y=1}^{c} \theta_y p(\boldsymbol{x}|y), \tag{4}$$

where θ_y is a coefficient corresponding to $p'(y)$:

$$\sum_{y=1}^{c} \theta_y = 1. \tag{5}$$

We match the model $q'(\boldsymbol{x})$ with the test input density $p'(\boldsymbol{x})$ under the KL divergence:

$$\begin{aligned}\mathrm{KL}(p'\|q') &:= \int p'(\boldsymbol{x}) \log \frac{p'(\boldsymbol{x})}{q'(\boldsymbol{x})} \mathrm{d}\boldsymbol{x} \\ &= \int p'(\boldsymbol{x}) \log p'(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \\ &\quad - \int p'(\boldsymbol{x}) \log \left(\sum_{y=1}^{c} \theta_y p(\boldsymbol{x}|y) \right) \mathrm{d}\boldsymbol{x}. \qquad (6)\end{aligned}$$

Ignoring the first term (which is a constant) and approximating the expectation in the second term with its empirical average give the following optimization problem:

$$\max_{\{\theta_y\}_{y=1}^{c}} \frac{1}{n'} \sum_{i=1}^{n'} \log \left(\sum_{y=1}^{c} \theta_y p(\boldsymbol{x}'_i|y) \right), \qquad (7)$$

subject to Eq.(5).

Since the above maximization is a convex optimization problem, the Karush-Kuhn-Tucker (KKT) conditions are necessary and sufficient for optimality (Boyd & Vandenberghe, 2004). The KKT conditions for the above problem is given by Eq.(5) and

$$\frac{1}{n'} \sum_{i=1}^{n'} \frac{p(\boldsymbol{x}'_i|y)}{\sum_{y'=1}^{c} \theta_{y'} p(\boldsymbol{x}'_i|y')} = \nu, \quad \forall y = 1, \ldots, c,$$

where ν is a Lagrange multiplier. From these equations, we can determine ν as

$$\begin{aligned}\nu = 1 \cdot \nu &= \left(\sum_{y=1}^{c} \theta_y \right) \cdot \left(\frac{1}{n'} \sum_{i=1}^{n'} \frac{p(\boldsymbol{x}'_i|y)}{\sum_{y'=1}^{c} \theta_{y'} p(\boldsymbol{x}'_i|y')} \right) \\ &= \frac{1}{n'} \sum_{i=1}^{n'} \frac{\sum_{y=1}^{c} \theta_y p(\boldsymbol{x}'_i|y)}{\sum_{y'=1}^{c} \theta_{y'} p(\boldsymbol{x}'_i|y')} = 1.\end{aligned}$$

Then the solution $\{\theta_y\}_{y=1}^{c}$ can be calculated by fixed-point iteration as follows (McLachlan & Krishnan, 1997):

$$\theta_y \longleftarrow \theta_y \left(\frac{1}{n'} \sum_{i=1}^{n'} \frac{p(\boldsymbol{x}'_i|y)}{\sum_{y=1}^{c} \theta_y p(\boldsymbol{x}'_i|y)} \right). \qquad (8)$$

Making the substitution $p(\boldsymbol{x}'_i|y) = p(y|\boldsymbol{x}'_i)p(\boldsymbol{x}'_i)/p(y)$, canceling $p(\boldsymbol{x}'_i)$ in the numerator and denominator, and replacing $p(y|\boldsymbol{x})$ with $\widehat{p}(y|\boldsymbol{x})$, we can show that the above updating formula is reduced to

$$\theta_y \longleftarrow \frac{1}{n'} \sum_{i=1}^{n'} \frac{\theta_y \widehat{p}(y|\boldsymbol{x}'_i)/\widehat{p}(y)}{\sum_{y'=1}^{c} \theta_{y'} \widehat{p}(y'|\boldsymbol{x}'_i)/\widehat{p}(y')},$$

which is the same as Eq.(3) with Eq.(2) substituted.

Therefore, the EM method is essentially equivalent to matching the training and test input distributions under the KL divergence, which uses the class-conditional density $p(\boldsymbol{x}|y)$ as a building block (see Eq.(8)). However, this fact is not apparent in the EM expression because of the cancellation of $p(\boldsymbol{x}'_i)$ in the numerator and denominator.

The convexity of Eq.(7) implies that there are no local minima. However, this was not recognized in Saerens et al. (2001) since the algorithm was derived via the incomplete data EM method.

4. Class-Prior Estimation by Direct Divergence Minimization

The analysis in the previous section motivates us to explore a more direct way to learn coefficients $\{\theta_y\}_{y=1}^{c}$. That is, given an estimator of a divergence from p' to q', coefficients $\{\theta_y\}_{y=1}^{c}$ are learned so that the divergence estimator is minimized.

In this section, we first review a general framework of approximating the f-*divergences* (Ali & Silvey, 1966; Csiszár, 1967) via *Legendre-Fenchel convex duality* (Keziou, 2003; Nguyen et al., 2010). Then we review two specific methods of divergence estimation for the KL divergence and the Pearson (PE) divergence (Pearson, 1900). Finally, we propose to use the PE divergence estimator for determining the coefficients $\{\theta_y\}_{y=1}^{c}$.

4.1. Framework of f-Divergence Approximation

An f-divergence (Ali & Silvey, 1966; Csiszár, 1967) from p' to q' is a general divergence measure defined by a convex function f such that $f(1) = 0$ as

$$D_f(p'\|q') := \int p'(\boldsymbol{x}) f\left(\frac{q'(\boldsymbol{x})}{p'(\boldsymbol{x})} \right) \mathrm{d}\boldsymbol{x}.$$

It was shown that the f-divergence can be lower-bounded via *Legendre-Fenchel convex duality* (Rockafellar, 1970) as follows (Keziou, 2003; Nguyen et al., 2010):

$$\begin{aligned}D_f(p'\|q') = \max_{r} \Bigg[&\int q'(\boldsymbol{x}) r(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \\ &- \int p'(\boldsymbol{x}) f^*(r(\boldsymbol{x})) \mathrm{d}\boldsymbol{x} \Bigg], \qquad (9)\end{aligned}$$

where f^* is the convex conjugate of f. The maximum is achieved if and only if $r(\boldsymbol{x}) = q'(\boldsymbol{x})/p'(\boldsymbol{x})$. Eq.(9) is a useful expression because the right-hand side only contains expectations of r and $f^*(r(\boldsymbol{x}))$, which can be simply approximated by sample averages.

Below, we show specific methods of divergence approximation for the KL and PE divergences under model (4) and the following parametric expression of the density ratio $r(\boldsymbol{x})$:

$$r(\boldsymbol{x}) = \sum_{\ell=0}^{b} \alpha_\ell \varphi_\ell(\boldsymbol{x}), \tag{10}$$

where $\{\alpha_\ell\}_{\ell=0}^{b}$ are parameters and $\{\varphi_\ell(\boldsymbol{x})\}_{\ell=0}^{b}$ are basis functions. In practice, we use a constant basis and Gaussian kernels centered at the training data points, i.e., for $b = n$ and $\ell = 1, 2, \ldots, n$,

$$\varphi_0(\boldsymbol{x}) = 1 \quad \text{and} \quad \varphi_\ell(\boldsymbol{x}) = \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{x}_\ell\|^2}{2\sigma^2}\right).$$

This provides a non-parametric divergence estimator (Nguyen et al., 2010; Sugiyama et al., 2008; Kanamori et al., 2012).

4.2. KL-Divergence Approximation

With $f(u) = -\log u$ for $u > 0$ and $+\infty$ for $u \leq 0$, the f-divergence is reduced to the KL divergence. For this f, the convex conjugate is given by $f^*(v) = -1 - \log(-v)$ for $v < 0$ and $+\infty$ for $v \geq 0$. Then, if $-\alpha_\ell$ is regarded as α_ℓ, an empirical approximation of Eq.(9) under (4) and (10) is given as follows (Nguyen et al., 2010):

$$\mathrm{KL}(p'\|q') \approx \max_{\{\alpha_\ell\}_{\ell=0}^{b}} \Bigg[-\sum_{y=1}^{c} \frac{\theta_y}{n_y} \sum_{i:y_i=y} \sum_{\ell=0}^{b} \alpha_\ell \varphi_\ell(\boldsymbol{x}_i) + \frac{1}{n'} \sum_{i=1}^{n'} \log\left(\sum_{\ell=0}^{b} \alpha_\ell \varphi_\ell(\boldsymbol{x}'_i)\right) + 1 \Bigg],$$

subject to $\alpha_0, \alpha_1, \ldots, \alpha_b \geq 0$. A similar approach, which directly estimates the inverted ratio $p'(\boldsymbol{x})/q'(\boldsymbol{x})$ with the same model (10), is also known (Sugiyama et al., 2008):

$$\mathrm{KL}(p'\|q') \approx \max_{\{\alpha_\ell\}_{\ell=0}^{b}} \left[\frac{1}{n'} \sum_{i=1}^{n'} \log\left(\sum_{\ell=0}^{b} \alpha_\ell \varphi_\ell(\boldsymbol{x}'_i)\right) \right],$$

subject to $\alpha_0, \alpha_1, \ldots, \alpha_b \geq 0$ and

$$\sum_{y=1}^{c} \frac{\theta_y}{n_y} \sum_{i:y_i=y} \sum_{\ell=0}^{b} \alpha_\ell \varphi_\ell(\boldsymbol{x}_i) = 1.$$

These are convex optimization problems, and thus global optimal solutions can be obtained by naive optimization. Tuning parameters possibly included in the basis function such as the kernel width can be systematically optimized by cross-validation (Sugiyama et al., 2008). The KL-divergence estimator obtained above was proved to possess superior convergence properties both in parametric and non-parametric setups (Sugiyama et al., 2008; Nguyen et al., 2010).

However, computing the KL-divergence estimator is rather time-consuming because optimization of $\{\alpha_\ell\}_{\ell=0}^{b}$ needs to be carried out for each $\{\theta_y\}_{y=1}^{c}$.

4.3. PE-Divergence Approximation

As an alternative to the KL-divergence, let us consider the PE divergence defined by

$$\mathrm{PE}(p'\|q') := \frac{1}{2} \int \left(\frac{q'(\boldsymbol{x})}{p'(\boldsymbol{x})} - 1\right)^2 p'(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}, \tag{11}$$

which is a squared-loss variant of the KL divergence and is a f-divergence with $f(u) = (t-1)^2/2$.

For this f, the convex conjugate is given by $f^*(v) = v^2/2 + v$. Then, an empirical approximation of Eq.(9) under (4) and (10) is given as follows (Kanamori et al., 2009a):

$$\mathrm{PE}(p'\|q') \approx \max_{\boldsymbol{\alpha}} \left[-\frac{1}{2} \boldsymbol{\alpha}^\top \widehat{\boldsymbol{G}} \boldsymbol{\alpha} + \boldsymbol{\alpha}^\top \widehat{\boldsymbol{H}} \boldsymbol{\theta} - \frac{1}{2} \right],$$

where

$$\boldsymbol{\alpha} = [\alpha_0 \ \alpha_1 \ \cdots \ \alpha_b]^\top, \quad \widehat{\boldsymbol{G}} = \frac{1}{n'} \sum_{i=1}^{n'} \boldsymbol{\varphi}(\boldsymbol{x}'_i) \boldsymbol{\varphi}(\boldsymbol{x}'_i)^\top,$$

$$\boldsymbol{\varphi}(\boldsymbol{x}) = [\varphi_0(\boldsymbol{x}) \ \varphi_1(\boldsymbol{x}) \ \cdots \ \varphi_b(\boldsymbol{x})], \ \widehat{\boldsymbol{H}} = \left[\widehat{\boldsymbol{h}}_1 \ \cdots \ \widehat{\boldsymbol{h}}_c\right],$$

$$\widehat{\boldsymbol{h}}_y = \frac{1}{n_y} \sum_{i:y_i=y} \boldsymbol{\varphi}(\boldsymbol{x}_i), \quad \boldsymbol{\theta} = [\theta_1 \ \theta_2 \ \cdots \ \theta_c]^\top.$$

A regularized solution to the above maximization problem can be obtained analytically as

$$\widehat{\boldsymbol{\alpha}} = \left(\widehat{\boldsymbol{G}} + \lambda \boldsymbol{R}\right)^{-1} \widehat{\boldsymbol{H}} \boldsymbol{\theta}, \tag{12}$$

where λ is a positive constant and $\boldsymbol{R}$ is defined as

$$\boldsymbol{R} = \begin{bmatrix} 0 & \boldsymbol{0}_{1\times b} \\ \boldsymbol{0}_{b\times 1} & \boldsymbol{I}_{b\times b} \end{bmatrix}.$$

The PE divergence estimator obtained above was proved to have superior convergence properties both in parametric and non-parametric setups (Kanamori et al., 2009a; 2012). Tuning parameters possibly included in the basis function such as the kernel width or the regularization parameter can be systematically optimized by cross-validation (Kanamori et al., 2009a; 2012).

4.4. Learning Class Ratios by PE Divergence Matching

As shown above, the KL and PE divergences can be systematically estimated without density estimation via Legendre-Fenchel convex duality. Among them, the PE divergence estimator, explicitly expressed as

$$\widehat{\mathrm{PE}}(\boldsymbol{\theta}) := -\frac{1}{2} \boldsymbol{\theta}^\top \widehat{\boldsymbol{H}}^\top \left(\widehat{\boldsymbol{G}} + \lambda \boldsymbol{R}\right)^{-1} \widehat{\boldsymbol{G}} \left(\widehat{\boldsymbol{G}} + \lambda \boldsymbol{R}\right)^{-1} \widehat{\boldsymbol{H}} \boldsymbol{\theta} + \boldsymbol{\theta}^\top \widehat{\boldsymbol{H}}^\top \left(\widehat{\boldsymbol{G}} + \lambda \boldsymbol{R}\right)^{-1} \widehat{\boldsymbol{H}} \boldsymbol{\theta} - \frac{1}{2},$$

is more useful for our purpose of learning class ratios, because of the following reasons: The PE-divergence was shown to be more robust against outliers than the KL-divergence, based on power divergence analysis (Basu et al., 1998; Sugiyama et al., 2012). This is a useful property in practical data analysis suffering high noise and outliers. Furthermore, the above PE-divergence estimator was shown to possess the minimum condition number among a general class of estimators, meaning that it is the most stable estimator (Kanamori et al., 2009b).

Another, and practically more important advantage of the above PE divergence estimator is that it can be computed efficiently and analytically. This advantage is even more crucial in our case because we minimize the above PE divergence estimator with respect to $\boldsymbol{\theta}$:

$$\min_{\boldsymbol{\theta}} \widehat{\mathrm{PE}}(\boldsymbol{\theta})$$

$$\text{subject to } \sum_{y=1}^{c} \theta_y = 1 \text{ and } \theta_1, \ldots, \theta_c \geq 0.$$

Because $\widehat{\mathrm{PE}}(\boldsymbol{\theta})$ is given analytically as a function of $\boldsymbol{\theta}$, we can easily obtain the minimizer $\widehat{\boldsymbol{\theta}}$ by simple optimization strategies such as alternate gradient descent and projection or just a grid search, without re-computing the PE divergence estimator.

5. Experiments

In this section, we report experimental results.

5.1. Setup

The following five methods are compared:

- **EM-KLR**: The method of Saerens et al. (2001) (see Section 2.2). The class-posterior probability of the training dataset is estimated using ℓ_2-penalized kernel logistic regression with Gaussian kernels. The L-BFGS quasi-Newton implementation included in the 'minFunc' package is used for logistic regression training (Schmidt, 2005).
- **KL-KDE**: The KL divergence estimator based on kernel density estimation (KDE). The class-wise input densities are estimated by KDE with Gaussian kernels. The kernel widths are estimated using likelihood cross-validation (Silverman, 1986).
- **PE-KDE**: The PE divergence estimator based on KDE. The class-wise input densities are estimated by KDE with Gaussian kernels. The kernel widths are estimated using least-squares cross-validation (Silverman, 1986).

Table 1. Datasets used in the experiments.

Dataset	d	# samples	# positives	# negatives
Australian	14	690	307	383
Diabetes	8	768	500	268
German	24	1000	300	700
Ionosphere	34	351	225	126
SAHeart	9	462	302	160
Twonorm	20	7400	3697	3703

- **KL-DR**: The proposed method (see Section 4.2) using a KL divergence estimator based on the density ratio (DR). For the optimization, the L-BFGS with projection implementation 'minFuncBC' is used (Schmidt, 2005).
- **PE-DR**: The proposed method (see Section 4.4) using the PE divergence estimator based on DR.

Below, we compare accuracy of class-prior estimation and classification.

5.2. Benchmark Datasets

Here, we use binary-classification benchmark datasets listed in Table 1. We select 10 samples from each of the two classes for the training dataset and 50 samples for the test dataset. The samples in the test set are selected with probability θ^* from the first class and $(1 - \theta^*)$ from the second class, where $\theta^* = 0.1, 0.2, 0.3, 0.4, 0.5$.

The average squared error of the estimated class ratios are given in Figure 1. This shows that methods based on the KL and PE divergences overall outperform EM-KLR, implying that our reformulation of the EM algorithm as distribution matching (see Section 3) contributes to obtaining accurate class-ratio estimates. Among the KL-based methods, KL-KDE tends to perform better than KL-DR. This is because, in KL-KDE, we did not estimate the first term in Eq.(6), which is the negative entropy and is a constant. On the other hand, the negative entropy is also implicitly estimated in KL-DR, possibly incurring additional estimation error. Among the PE-based methods, PE-DR outperforms PE-KDE, showing that directly estimating density ratios without density estimation is more promising as a PE divergence estimator. Overall, PE-DR is shown to be the most accurate.

Next, we compare classification accuracy when the learned class-prior probabilities are used as instance weights. Figure 2 shows misclassification rates for a regularized least-squares classifier (Rifkin et al., 2003) with instance weighting. The results show that, as expected, a more accurate estimate of the class ratio tends to give a lower misclassification rate.

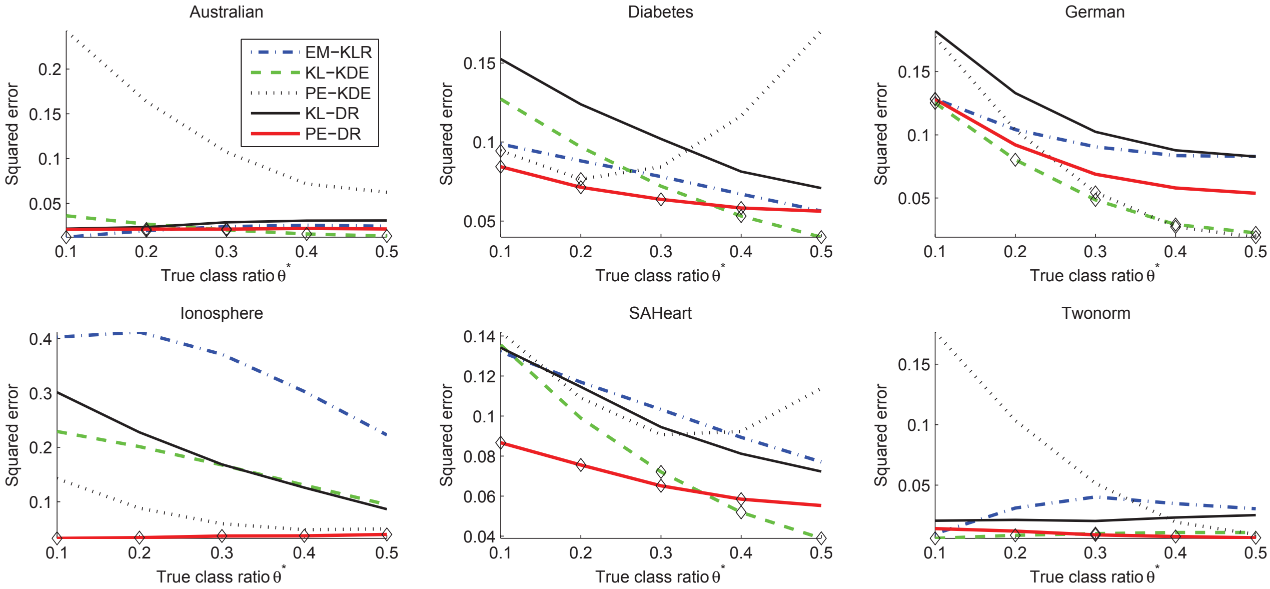


Figure 1. Average squared error between the true class ratio θ^* and estimated class ratio $\widehat{\theta}$ for the benchmark datasets listed in Table 1. The best method and comparable methods according to the t-test at significance level of 5% are indicated with a '$\diamond$'

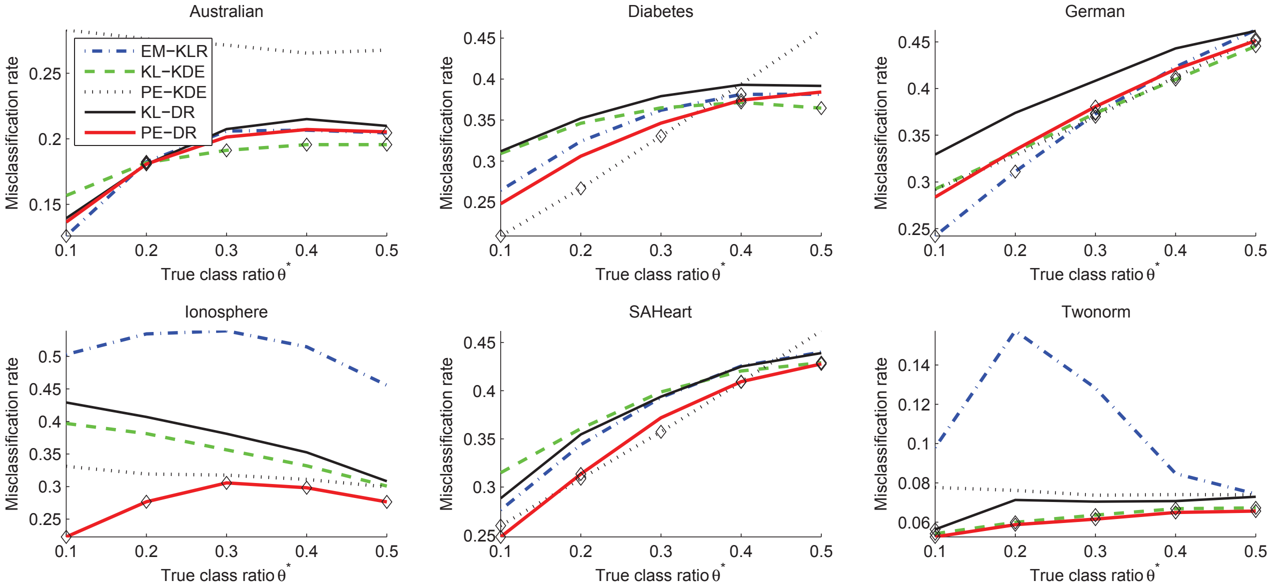


Figure 2. Average misclassification rates for the datasets listed in Table 1. Classification is performed using a regularized least-squares classifier with instance weighting. The best method and comparable methods according to the t-test at significance level of 5% are indicated with a '$\diamond$'.

5.3. Real-World Application

Finally, we demonstrate the usefulness of the proposed approach in a real-world problem of military vehicle classification from geophone recordings (Duarte & Hu, 2004). This is a three class problem: Two vehicle classes and a class of recorded noise. The features are 50-dimensional. In this vehicle classification task, class-prior change is inevitable because the type of vehicles passing through differs depending on time (e.g., day and night).

n samples are drawn from each of the labeled classes for the training set with the uniform class prior, whereas 100 samples are drawn with probabilities $p = [0.6\ 0.1\ 0.3]$ from each of the classes for the test set. Due to the prohibitive computational cost, KL-DR was not included in this experiment.

In Figure 3, we plot the ℓ_2-distance between the true and estimated class priors and the misclassification rate based on instance-weighted kernel logistic regression (Hastie et al., 2001) averaged over 1000 runs as functions of the number of training samples. As can be seen from the graphs, the performance of all methods improves as the number of training samples increases. Among the compared methods, PE-DR provides the most accurate estimates of the class prior and thus yields the lowest classification error.

6. Conclusion

Class-prior change is a problem that is conceivable in many real-world datasets, and it can be systematically corrected for if the class-prior of the test dataset is known. In this paper, we discussed the problem of estimating the test class ratios under the semi-supervised learning setup.

We first showed that the EM-based estimator introduced in Saerens et al. (2001) can be regarded as indirectly matching the test input distribution by a linear combination of class-wise input distributions. Based on this view, we proposed to use an explicit and possibly more accurate divergence estimator based on density-ratio estimation (Kanamori et al., 2009a) for learning test class-priors. The proposed method was shown to have various nice properties such as high robustness to noise and outliers, superior numerical stability, and excellent computational efficiency. Through experiments, we showed that the class ratios estimated by the proposed method are more accurate than competing methods, which can be translated into better classification accuracy.

Acknowledgments

The authors thank the anonymous reviewers for their helpful comments. MCdP was supported by the MEXT scholarship, and MS was supported by AOARD and JST PRESTO.

References

Ali, S. M. and Silvey, S. D. A general class of coefficients of divergence of one distribution from another. *Journal of the Royal Statistical Society, Series B*, 28:131–142, 1966.

Basu, A., Harris, I. R., Hjort, N. L., and Jones, M. C. Robust and efficient estimation by minimising a density power divergence. *Biometrika*, 85(3):549–559, 1998.

Bishop, C. M. *Pattern Recognition and Machine Learning*. Springer, New York, NY, USA, 2006.

Boyd, S. and Vandenberghe, L. *Convex Optimization*. Cambridge University Press, New York, NY, USA, 2004.

Chan, Y. S. and Ng, H. T. Estimating class priors in domain adaptation for word sense disambiguation. In *Proceedings of the 21st International Conference on Computational Linguistics*, pp. 89–96, 2006.

Chapelle, O., Schölkopf, B., and Zien, A. (eds.). *Semi-Supervised Learning*. MIT Press, Cambridge, MA, USA, 2006.

Clémençon, S., Vayatis, N., and Depecker, M. AUC optimization and the two-sample problem. In *Advances in Neural Information Processing Systems 22*, pp. 360–368. 2009.

Cortes, C. and Mohri, M. AUC optimization vs. error rate minimization. In *Advances in Neural Information Processing Systems 16*. MIT Press, Cambridge, MA, 2004.

Csiszár, I. Information-type measures of difference of probability distributions and indirect observation. *Studia Scientiarum Mathematicarum Hungarica*, 2:229–318, 1967.

Dempster, A. P., Laird, N. M., and Rubin, D. B. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, series B*, 39(1): 1–38, 1977.

Duarte, M. F. and Hu, Y. H. Vehicle classification in distributed sensor networks. *Journal of Parallel and Distributed Computing*, 64(7):826–838, 2004.

Elkan, C. The foundations of cost-sensitive learning. In *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence*, pp. 973–978, 2001.

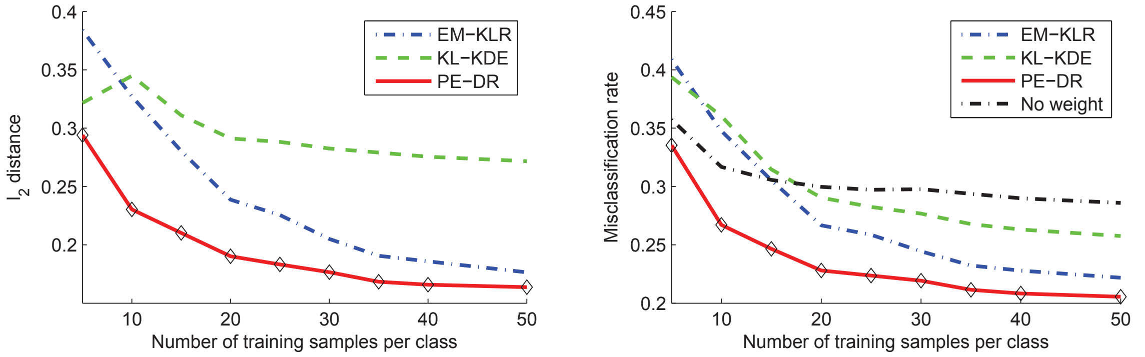
Hastie, T., Tibshirani, R., and Friedman, J. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer, New York, NY, USA, 2001.

Heckman, J. J. Sample selection bias as a specification error. *Econometrica*, 47(1):153–161, 1979.

Kanamori, T., Hido, S., and Sugiyama, M. A least-squares approach to direct importance estimation. *Journal of Machine Learning Research*, 10:1391–1445, 2009a.

Kanamori, T., Suzuki, T., and Sugiyama, M. Condition number analysis of kernel-based density ratio estimation. Technical report, arXiv, 2009b.

Kanamori, T., Suzuki, T., and Sugiyama, M. Statistical analysis of kernel-based least-squares density-ratio estimation. *Machine Learning*, 2012.



(a) The ℓ_2-distance between the true and estimated class priors.

(b) Misclassification rate with instance-weighted kernel logistic regression.

Figure 3. Experimental results for the vehicle classification problem. The best method and comparable methods according to the t-test at significance level of 5% are indicated with a '$\diamond$'.

Keziou, A. Dual representation of ϕ-divergences and applications. *Comptes Rendus Mathématique*, 336(10):857–862, 2003.

Kullback, S. and Leibler, R. A. On information and sufficiency. *Annals of Mathematical Statistics*, 22:79–86, 1951.

Latinne, P., Saerens, M., and Decaestecker, C. Adjusting the outputs of a classifier to new a priori probabilities may significantly improve classification accuracy: Evidence from a multi-class problem in remote sensing. In *Proceedings of the 18th International Conference on Machine Learning*, pp. 298–305, 2001.

Lin, Y., Lee, Y., and Wahba, G. Support vector machines for classification in nonstandard situations. *Machine Learning*, 46(1/3):191–202, 2002.

McLachlan, G. J. and Krishnan, T. *The EM algorithm and extensions*. Wiley series in probability and statistics: Applied probability and statistics. John Wiley, 1997.

Nguyen, X., Wainwright, M. J., and Jordan, M. I. Estimating divergence functionals and the likelihood ratio by convex risk minimization. *IEEE Transactions on Information Theory*, 56(11):5847–5861, 2010.

Pearson, K. On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling. *Philosophical Magazine*, 50:157–175, 1900.

Quiñonero-Candela, J., Sugiyama, M., Schwaighofer, A., and Lawrence, N. (eds.). *Dataset Shift in Machine Learning*. MIT Press, Cambridge, MA, USA, 2009.

Rifkin, R., Yeo, G., and Poggio, T. Regularized least-squares classification. *Advances in Learning Theory: Methods, Model and Applications. NATO Science Series III: Computer and Systems Sciences*, 190:131–153, 2003.

Rockafellar, R. T. *Convex Analysis*. Princeton University Press, Princeton, NJ, USA, 1970.

Saerens, M., Patrice, M., and Decaestecker, C. Adjusting the outputs of a classifier to new a priori probabilities: A simple procedure. *Neural Computation*, 14:21–41, 2001.

Schmidt, M. minFunc—Unconstrained differentiable multivariate optimization in MATLAB, 2005.

Silverman, B. W. *Density Estimation: For Statistics and Data Analysis*. Chapman and Hall, London, UK, 1986.

Sugiyama, M. Superfast-trainable multi-class probabilistic classifier by least-squares posterior fitting. *IEICE Transactions on Information and Systems*, E93-D:2690–2701, 2010.

Sugiyama, M. and Kawanabe, M. *Machine Learning in Non-Stationary Environments: Introduction to Covariate Shift Adaptation*. MIT Press, Cambridge, MA, USA, 2012.

Sugiyama, M., Suzuki, T., Nakajima, S., Kashima, H., von Bünau, P., and Kawanabe, M. Direct importance estimation for covariate shift adaptation. *Annals of the Institute of Statistical Mathematics*, 60(4):699–746, 2008.

Sugiyama, M., Suzuki, T., and Kanamori, T. Density ratio matching under the Bregman divergence: A unified framework of density ratio estimation. *Annals of the Institute of Statistical Mathematics*, 2012.

Vapnik, V. N. *Statistical Learning Theory*. Wiley, New York, NY, USA, 1998.

Density-Difference Estimation

Masashi Sugiyama[1] **Takafumi Kanamori**[2] **Taiji Suzuki**[3]
Marthinus Christoffel du Plessis[1] **Song Liu**[1] **Ichiro Takeuchi**[4]
[1]Tokyo Institute of Technology, Japan [2]Nagoya University, Japan
[3]University of Tokyo, Japan [4]Nagoya Institute of Technology, Japan

Abstract

We address the problem of estimating the *difference* between two probability densities. A naive approach is a two-step procedure of first estimating two densities separately and then computing their difference. However, such a two-step procedure does not necessarily work well because the first step is performed without regard to the second step and thus a small estimation error incurred in the first stage can cause a big error in the second stage. In this paper, we propose a single-shot procedure for directly estimating the density difference without separately estimating two densities. We derive a non-parametric finite-sample error bound for the proposed single-shot density-difference estimator and show that it achieves the optimal convergence rate. We then show how the proposed density-difference estimator can be utilized in L^2-distance approximation. Finally, we experimentally demonstrate the usefulness of the proposed method in robust distribution comparison such as class-prior estimation and change-point detection.

1 Introduction

When estimating a quantity consisting of two elements, a two-stage approach of first estimating the two elements separately and then approximating the target quantity based on the estimates of the two elements often performs poorly, because the first stage is carried out without regard to the second stage and thus a small estimation error incurred in the first stage can cause a big error in the second stage. To cope with this problem, it would be more appropriate to directly estimate the target quantity in a single-shot process without separately estimating the two elements.

A seminal example that follows this general idea is pattern recognition by the *support vector machine* [1]: Instead of separately estimating two probability distributions of patterns for positive and negative classes, the support vector machine directly learns the boundary between the two classes that is sufficient for pattern recognition. More recently, a problem of estimating the ratio of two probability densities was tackled in a similar fashion [2, 3]: The ratio of two probability densities is directly estimated without going through separate estimation of the two probability densities.

In this paper, we further explore this line of research, and propose a method for directly estimating the *difference* between two probability densities in a single-shot process. Density differences would be more desirable than density ratios because density ratios can diverge to infinity even under a mild condition (e.g., two Gaussians [4]), whereas density differences are always finite as long as each density is bounded. Density differences can be used for solving various machine learning tasks such as class-balance estimation under class-prior change [5] and change-point detection in time series [6].

For this density-difference estimation problem, we propose a single-shot method, called the *least-squares density-difference* (LSDD) estimator, that directly estimates the density difference without separately estimating two densities. LSDD is derived with in the framework of kernel regularized least-squares estimation, and thus it inherits various useful properties: For example, the LSDD solution can be computed *analytically* in a computationally efficient and stable manner, and all tuning parameters such as the kernel width and the regularization parameter can be systematically and objectively optimized via cross-validation. We derive a finite-sample error bound for the LSDD estimator and show that it achieves the optimal convergence rate in a non-parametric setup.

We then apply LSDD to L^2-distance estimation and show that it is more accurate than the difference of KDEs, which tends to severely under-estimate the L^2-distance [7]. Because the L^2-distance is more robust against outliers than the *Kullback-Leibler divergence* [8], the proposed L^2-distance estimator can lead to the paradigm of robust distribution comparison. We experimentally demonstrate the usefulness of LSDD in semi-supervised class-prior estimation and unsupervised change detection.

2 Density-Difference Estimation

In this section, we propose a single-shot method for estimating the difference between two probability densities from samples, and analyze its theoretical properties.

Problem Formulation and Naive Approach: First, we formulate the problem of density-difference estimation. Suppose that we are given two sets of independent and identically distributed samples $\mathcal{X} := \{\boldsymbol{x}_i\}_{i=1}^{n}$ and $\mathcal{X}' := \{\boldsymbol{x}'_{i'}\}_{i'=1}^{n'}$ from probability distributions on $\mathbb{R}^d$ with densities $p(\boldsymbol{x})$ and $p'(\boldsymbol{x})$, respectively. Our goal is to estimate the density difference,

$$f(\boldsymbol{x}) := p(\boldsymbol{x}) - p'(\boldsymbol{x}),$$

from the samples $\mathcal{X}$ and $\mathcal{X}'$.

A naive approach to density-difference estimation is to use *kernel density estimators* (KDEs). However, we argue that the KDE-based density-difference estimator is not the best approach because of its two-step nature. Intuitively, good density estimators tend to be smooth and thus the difference between such smooth density estimators tends to be over-smoothed as a density-difference estimator [9]. To overcome this weakness, we give a single-shot procedure of directly estimating the density difference $f(\boldsymbol{x})$ without separately estimating the densities $p(\boldsymbol{x})$ and $p'(\boldsymbol{x})$.

Least-Squares Density-Difference Estimation: In our proposed approach, we fit a density-difference model $g(\boldsymbol{x})$ to the true density-difference function $f(\boldsymbol{x})$ under the squared loss:

$$\operatorname*{argmin}_{g} \int \Big(g(\boldsymbol{x}) - f(\boldsymbol{x})\Big)^2 \mathrm{d}\boldsymbol{x}.$$

We use the following Gaussian kernel model as $g(\boldsymbol{x})$:

$$g(\boldsymbol{x}) = \sum_{\ell=1}^{n+n'} \theta_\ell \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{c}_\ell\|^2}{2\sigma^2}\right), \tag{1}$$

where $(\boldsymbol{c}_1, \ldots, \boldsymbol{c}_n, \boldsymbol{c}_{n+1}, \ldots, \boldsymbol{c}_{n+n'}) := (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n, \boldsymbol{x}'_1, \ldots, \boldsymbol{x}'_{n'})$ are Gaussian kernel centers. If $n + n'$ is large, we may use only a subset of $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n, \boldsymbol{x}'_1, \ldots, \boldsymbol{x}'_{n'}\}$ as Gaussian kernel centers.

For the model (1), the optimal parameter $\boldsymbol{\theta}^*$ is given by

$$\boldsymbol{\theta}^* := \operatorname*{argmin}_{\boldsymbol{\theta}} \int \Big(g(\boldsymbol{x}) - f(\boldsymbol{x})\Big)^2 \mathrm{d}\boldsymbol{x} = \operatorname*{argmin}_{\boldsymbol{\theta}} \Big[\boldsymbol{\theta}^\top \boldsymbol{H}\boldsymbol{\theta} - 2\boldsymbol{h}^\top\boldsymbol{\theta}\Big] = \boldsymbol{H}^{-1}\boldsymbol{h},$$

where $\boldsymbol{H}$ is the $(n+n') \times (n+n')$ matrix and $\boldsymbol{h}$ is the $(n+n')$-dimensional vector defined as

$$H_{\ell,\ell'} := \int \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{c}_\ell\|^2}{2\sigma^2}\right) \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{c}_{\ell'}\|^2}{2\sigma^2}\right) \mathrm{d}\boldsymbol{x} = (\pi\sigma^2)^{d/2} \exp\left(-\frac{\|\boldsymbol{c}_\ell - \boldsymbol{c}_{\ell'}\|^2}{4\sigma^2}\right),$$

$$h_\ell := \int \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{c}_\ell\|^2}{2\sigma^2}\right) p(\boldsymbol{x})\mathrm{d}\boldsymbol{x} - \int \exp\left(-\frac{\|\boldsymbol{x}' - \boldsymbol{c}_\ell\|^2}{2\sigma^2}\right) p'(\boldsymbol{x}')\mathrm{d}\boldsymbol{x}'.$$

Replacing the expectations in $\boldsymbol{h}$ by empirical estimators and adding an ℓ_2-regularizer to the objective function, we arrive at the following optimization problem:

$$\widehat{\boldsymbol{\theta}} := \operatorname*{argmin}_{\boldsymbol{\theta}} \Big[\boldsymbol{\theta}^\top \boldsymbol{H}\boldsymbol{\theta} - 2\widehat{\boldsymbol{h}}^\top\boldsymbol{\theta} + \lambda\boldsymbol{\theta}^\top\boldsymbol{\theta}\Big], \tag{2}$$

where λ (≥ 0) is the regularization parameter and $\widehat{\boldsymbol{h}}$ is the $(n+n')$-dimensional vector defined as

$$\widehat{h}_\ell := \frac{1}{n}\sum_{i=1}^{n} \exp\left(-\frac{\|\boldsymbol{x}_i - \boldsymbol{c}_\ell\|^2}{2\sigma^2}\right) - \frac{1}{n'}\sum_{i'=1}^{n'} \exp\left(-\frac{\|\boldsymbol{x}'_{i'} - \boldsymbol{c}_\ell\|^2}{2\sigma^2}\right).$$

Taking the derivative of the objective function in Eq.(2) and equating it to zero, we can obtain the solution analytically as

$$\widehat{\boldsymbol{\theta}} = (\boldsymbol{H} + \lambda \boldsymbol{I})^{-1}\,\widehat{\boldsymbol{h}},$$

where $\boldsymbol{I}$ denotes the identity matrix.

Finally, a density-difference estimator $\widehat{f}(\boldsymbol{x})$, which we call the *least-squares density-difference* (LSDD) estimator, is given as

$$\widehat{f}(\boldsymbol{x}) = \sum_{\ell=1}^{n+n'} \widehat{\theta}_\ell \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{c}_\ell\|^2}{2\sigma^2}\right).$$

Non-Parametric Error Bound: Here, we theoretically analyze an estimation error of LSDD.

We assume $n' = n$, and let $\mathcal{H}_\gamma$ be the reproducing kernel Hilbert space (RKHS) corresponding to the Gaussian kernel with width γ: $k_\gamma(\boldsymbol{x}, \boldsymbol{x}') = \exp\left(-\|\boldsymbol{x} - \boldsymbol{x}'\|^2/\gamma^2\right)$. Let us consider a slightly modified LSDD estimator that is more suitable for non-parametric error analysis[1]:

$$\widehat{f} := \mathop{\mathrm{argmin}}_{g \in \mathcal{H}_\gamma} \left[\|g\|^2_{L^2(\mathbb{R}^d)} - 2\left(\frac{1}{n}\sum_{i=1}^{n} g(\boldsymbol{x}_i) - \frac{1}{n}\sum_{i'=1}^{n} g(\boldsymbol{x}'_{i'})\right) + \lambda \|g\|^2_{\mathcal{H}_\gamma}\right].$$

Then we have the following theorem:

Theorem 1. *Suppose that there exists a constant M such that $\|p\|_\infty \leq M$ and $\|p'\|_\infty \leq M$. Suppose also that the density difference $f = p - p'$ is a member of Besov space with regularity α. That is, $f \in B^\alpha_{2,\infty}$ where $B^\alpha_{2,\infty}$ is the Besov space with regularity α, and*

$$\|f\|_{B^\alpha_{2,\infty}} := \|f\|_{L_2(\mathbb{R}^d)} + \sup_{t>0}(t^{-\alpha}\omega_{r,L_2(\mathbb{R}^d)}(f,t)) < c \ \ \textit{for}\ r = \lfloor \alpha \rfloor + 1,$$

where $\lfloor \alpha \rfloor$ denotes the largest integer less than or equal to α and $\omega_{r,L_2(\mathbb{R}^d)}$ is the r-th modulus of smoothness (see [10] for the definitions). Then, for all $\epsilon > 0$ and $p \in (0,1)$, there exists a constant $K > 0$ depending on M, c, ϵ, and p such that for all $n \geq 1$, $\tau \geq 1$, and $\lambda > 0$, the LSDD estimator $\widehat{f}$ in $\mathcal{H}_\gamma$ satisfies

$$\|\widehat{f} - f\|^2_{L^2(\mathbb{R}^d)} + \lambda \|\widehat{f}\|^2_{\mathcal{H}_\gamma} \leq K\left(\lambda\gamma^{-d} + \gamma^{2\alpha} + \frac{\gamma^{-(1-p)(1+\epsilon)d}}{\lambda^p n} + \frac{\gamma^{-\frac{2(1-p)d}{1+p}(1+\epsilon+\frac{1-p}{4})}}{\lambda^{\frac{3p-p^2}{1+p}} n^{\frac{2}{1+p}}} + \frac{\tau}{n^2\lambda} + \frac{\tau}{n}\right)$$

with probability not less than $1 - 4e^{-\tau}$.

If we set $\lambda = n^{-\frac{2\alpha+d}{(2\alpha+d)(1+p)+(\epsilon-p+\epsilon p)}}$ and $\gamma = n^{-\frac{1}{(2\alpha+d)(1+p)+(\epsilon-p+\epsilon p)}}$, and take ϵ and p sufficiently small, then we immediately have the following corollary.

Corollary 1. *Suppose that the same assumptions as Theorem 1 hold. Then, for all $\rho, \rho' > 0$, there exists a constant $K > 0$ depending on M, c, ρ, and ρ' such that, for all $n \geq 1$ and $\tau \geq 1$, the density-difference estimator $\widehat{f}$ with appropriate choice of γ and λ satisfies*

$$\|\widehat{f} - f\|^2_{L^2(\mathbb{R}^d)} + \lambda \|\widehat{f}\|^2_{\mathcal{H}_\gamma} \leq K\left(n^{-\frac{2\alpha}{2\alpha+d}+\rho} + \tau n^{-1+\rho'}\right)$$

with probability not less than $1 - 4e^{-\tau}$.

[1] More specifically, the regularizer is replaced from the squared ℓ_2-norm of parameters to the squared RKHS-norm of a learned function, which is necessary to establish consistency. Nevertheless, we use the squared ℓ_2-norm of parameters in experiments because it is simpler and seems to perform well in practice.

Note that $n^{-\frac{2\alpha}{2\alpha+d}}$ is the optimal learning rate to estimate a function in $B^{\alpha}_{2,\infty}$. Therefore, the density-difference estimator with a Gaussian kernel achieves the optimal learning rate by appropriately choosing the regularization parameter and the Gaussian width. Because the learning rate depends on α, the LSDD estimator has adaptivity to the smoothness of the true function.

It is known that, if the naive KDE with a Gaussian kernel is used for estimating a probability density with regularity $\alpha > 2$, the optimal learning rate cannot be achieved [11, 12]. To achieve the optimal rate by KDE, we should choose a kernel function specifically tailored to each regularity α [13]. However, such a kernel function is not non-negative and it is difficult to implement it in practice. On the other hand, our LSDD estimator can always achieve the optimal learning rate for a Gaussian kernel without regard to regularity α.

Model Selection by Cross-Validation: The above theoretical analysis showed the superiority of LSDD. However, in practice, the performance of LSDD depends on the choice of models (i.e., the kernel width σ and the regularization parameter λ). Here, we show that the model can be optimized by *cross-validation* (CV). More specifically, we first divide the samples $\mathcal{X} = \{\boldsymbol{x}_i\}_{i=1}^{n}$ and $\mathcal{X}' = \{\boldsymbol{x}'_{i'}\}_{i'=1}^{n'}$ into T disjoint subsets $\{\mathcal{X}_t\}_{t=1}^{T}$ and $\{\mathcal{X}'_t\}_{t=1}^{T}$, respectively. Then we obtain a density-difference estimate $\widehat{f}_t(\boldsymbol{x})$ from $\mathcal{X}\backslash\mathcal{X}_t$ and $\mathcal{X}'\backslash\mathcal{X}'_t$ (i.e., all samples without $\mathcal{X}_t$ and $\mathcal{X}'_t$), and compute its hold-out error for $\mathcal{X}_t$ and $\mathcal{X}'_t$ as

$$\mathrm{CV}^{(t)} := \int \widehat{f}_t(\boldsymbol{x})^2 \mathrm{d}\boldsymbol{x} - \frac{2}{|\mathcal{X}_t|}\sum_{\boldsymbol{x}\in\mathcal{X}_t}\widehat{f}_t(\boldsymbol{x}) + \frac{2}{|\mathcal{X}'_t|}\sum_{\boldsymbol{x}'\in\mathcal{X}'_t}\widehat{f}_t(\boldsymbol{x}'),$$

where $|\mathcal{X}|$ denotes the number of elements in the set $\mathcal{X}$. We repeat this hold-out validation procedure for $t = 1, \ldots, T$, and compute the average hold-out error. Finally, we choose the model that minimizes the average hold-out error.

3 L^2-Distance Estimation by LSDD

In this section, we consider the problem of approximating the L^2-distance between $p(\boldsymbol{x})$ and $p'(\boldsymbol{x})$,

$$L^2(p,p') := \int \left(p(\boldsymbol{x}) - p'(\boldsymbol{x})\right)^2 \mathrm{d}\boldsymbol{x},$$

from their independent and identically distributed samples $\mathcal{X} := \{\boldsymbol{x}_i\}_{i=1}^{n}$ and $\mathcal{X}' := \{\boldsymbol{x}'_{i'}\}_{i'=1}^{n'}$.

For an equivalent expression $L^2(p,p') = \int f(\boldsymbol{x})p(\boldsymbol{x})\mathrm{d}\boldsymbol{x} - \int f(\boldsymbol{x}')p'(\boldsymbol{x}')\mathrm{d}\boldsymbol{x}'$, if we replace $f(\boldsymbol{x})$ with an LSDD estimator $\widehat{f}(\boldsymbol{x})$ and approximate the expectations by empirical averages, we obtain $L^2(p,p') \approx \widehat{\boldsymbol{h}}^\top\widehat{\boldsymbol{\theta}}$. Similarly, for another expression $L^2(p,p') = \int f(\boldsymbol{x})^2\mathrm{d}\boldsymbol{x}$, replacing $f(\boldsymbol{x})$ with an LSDD estimator $\widehat{f}(\boldsymbol{x})$ gives $L^2(p,p') \approx \widehat{\boldsymbol{\theta}}^\top\boldsymbol{H}\widehat{\boldsymbol{\theta}}$.

Although $\widehat{\boldsymbol{h}}^\top\widehat{\boldsymbol{\theta}}$ and $\widehat{\boldsymbol{\theta}}^\top\boldsymbol{H}\widehat{\boldsymbol{\theta}}$ themselves give approximations to $L^2(p,p')$, we argue that the use of their combination, defined by

$$\widehat{L}^2(\mathcal{X},\mathcal{X}') := 2\widehat{\boldsymbol{h}}^\top\widehat{\boldsymbol{\theta}} - \widehat{\boldsymbol{\theta}}^\top\boldsymbol{H}\widehat{\boldsymbol{\theta}}, \tag{3}$$

is more sensible. To explain the reason, let us consider a generalized L^2-distance estimator of the form $\beta\widehat{\boldsymbol{h}}^\top\widehat{\boldsymbol{\theta}} + (1-\beta)\widehat{\boldsymbol{\theta}}^\top\boldsymbol{H}\widehat{\boldsymbol{\theta}}$, where β is a real scalar. If the regularization parameter λ (≥ 0) is small, this can be expressed as

$$\beta\widehat{\boldsymbol{h}}^\top\widehat{\boldsymbol{\theta}} + (1-\beta)\widehat{\boldsymbol{\theta}}^\top\boldsymbol{H}\widehat{\boldsymbol{\theta}} = \widehat{\boldsymbol{h}}^\top\boldsymbol{H}^{-1}\widehat{\boldsymbol{h}} - \lambda(2-\beta)\widehat{\boldsymbol{h}}^\top\boldsymbol{H}^{-2}\widehat{\boldsymbol{h}} + o_p(\lambda), \tag{4}$$

where o_p denotes the probabilistic order. Thus, up to $O_p(\lambda)$, the bias introduced by regularization (i.e., the second term in the right-hand side of Eq.(4) that depends on λ) can be eliminated if $\beta = 2$, which yields Eq.(3). Note that, if no regularization is imposed (i.e., $\lambda = 0$), both $\widehat{\boldsymbol{h}}^\top\widehat{\boldsymbol{\theta}}$ and $\widehat{\boldsymbol{\theta}}^\top\boldsymbol{H}\widehat{\boldsymbol{\theta}}$ yield $\widehat{\boldsymbol{h}}^\top\boldsymbol{H}^{-1}\widehat{\boldsymbol{h}}$, the first term in the right-hand side of Eq.(4).

Eq.(3) is actually equivalent to the negative of the optimal objective value of the LSDD optimization problem without regularization (i.e., Eq.(2) with $\lambda = 0$). This can be naturally interpreted through a lower bound of $L^2(p, p')$ obtained by *Legendre-Fenchel convex duality* [14]:

$$L^2(p,p') = \sup_{g} \left[2 \left(\int g(\boldsymbol{x}) p(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} - \int g(\boldsymbol{x}') p'(\boldsymbol{x}') \mathrm{d}\boldsymbol{x}' \right) - \int g(\boldsymbol{x})^2 \mathrm{d}\boldsymbol{x} \right],$$

where the supremum is attained at $g = f$. If the expectations are replaced by empirical estimators and the Gaussian kernel model (1) is used as g, the above optimization problem is reduced to the LSDD objective function without regularization (see Eq.(2)). Thus, LSDD corresponds to approximately maximizing the above lower bound and Eq.(3) is its maximum value.

Through eigenvalue decomposition of $\boldsymbol{H}$, we can show that $2\widehat{\boldsymbol{h}}^\top \widehat{\boldsymbol{\theta}} - \widehat{\boldsymbol{\theta}}^\top \boldsymbol{H} \widehat{\boldsymbol{\theta}} \geq \widehat{\boldsymbol{h}}^\top \widehat{\boldsymbol{\theta}} \geq \widehat{\boldsymbol{\theta}}^\top \boldsymbol{H} \widehat{\boldsymbol{\theta}}$. Thus, our approximator (3) is not less than the plain approximators $\widehat{\boldsymbol{h}}^\top \widehat{\boldsymbol{\theta}}$ and $\widehat{\boldsymbol{\theta}}^\top \boldsymbol{H} \widehat{\boldsymbol{\theta}}$.

4 Experiments

In this section, we experimentally demonstrate the usefulness of LSDD. A MATLAB® implementation of LSDD used for experiments is available from

"`http://sugiyama-www.cs.titech.ac.jp/~sugi/software/LSDD/`".

Illustration: Let $N(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ be the multi-dimensional normal density with mean vector $\boldsymbol{\mu}$ and variance-covariance matrix $\boldsymbol{\Sigma}$ with respect to $\boldsymbol{x}$, and let

$$p(\boldsymbol{x}) = N(\boldsymbol{x}; (\mu, 0, \ldots, 0)^\top, (4\pi)^{-1} \boldsymbol{I}_d) \quad \text{and} \quad p'(\boldsymbol{x}) = N(\boldsymbol{x}; (0, 0, \ldots, 0)^\top, (4\pi)^{-1} \boldsymbol{I}_d).$$

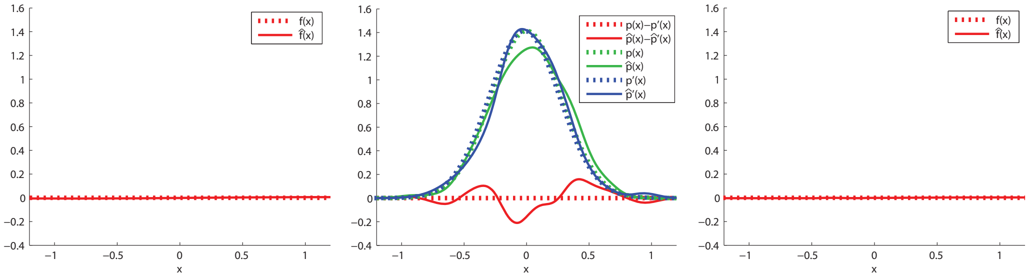
We first illustrate how LSDD behaves under $d = 1$ and $n = n' = 200$. We compare LSDD with KDEi (KDE with two Gaussian widths chosen *independently* by least-squares cross-validation [15]) and KDEj (KDE with two Gaussian widths chosen *jointly* to minimize the LSDD criterion [9]). The number of folds in cross-validation is set to 5 for all methods.

Figure 1 depicts density-difference estimation results obtained by LSDD, KDEi, and KDEj for $\mu = 0$ (i.e., $f(x) = p(x) - p'(x) = 0$). The figure shows that LSDD and KDEj give accurate estimates of the density difference $f(x) = 0$. On the other hand, the estimate obtained by KDEi is rather fluctuated, although both densities are reasonably well approximated by KDEs. This illustrates an advantage of directly estimating the density difference without going through separate estimation of each density. Figure 2 depicts the results for $\mu = 0.5$ (i.e., $f(x) \neq 0$), showing again that LSDD performs well. KDEi and KDEj give the same estimation result for this dataset, which slightly underestimates the peaks.

Next, we compare the performance of L^2-distance approximation based on LSDD, KDEi, and KDEj. For $\mu = 0, 0.2, 0.4, 0.6, 0.8$ and $d = 1, 5$, we draw $n = n' = 200$ samples from the above $p(\boldsymbol{x})$ and $p'(\boldsymbol{x})$. Figure 3 depicts the mean and standard error of estimated L^2-distances over 1000 runs as functions of mean μ. When $d = 1$ (Figure 3(a)), the LSDD-based L^2-distance estimator gives the most accurate estimates of the true L^2-distance, whereas the KDEi-based L^2-distance estimator slightly underestimates the true L^2-distance when μ is large. This is caused by the fact that KDE tends to provide smooth density estimates (see Figure 2(b) again): Such smooth density estimates are accurate as density estimates, but the difference of smooth density estimates yields a small L^2-distance estimate [7]. The KDEj-based L^2-distance estimator tends to improve this drawback of KDEi, but it still slightly underestimates the true L^2-distance when μ is large.

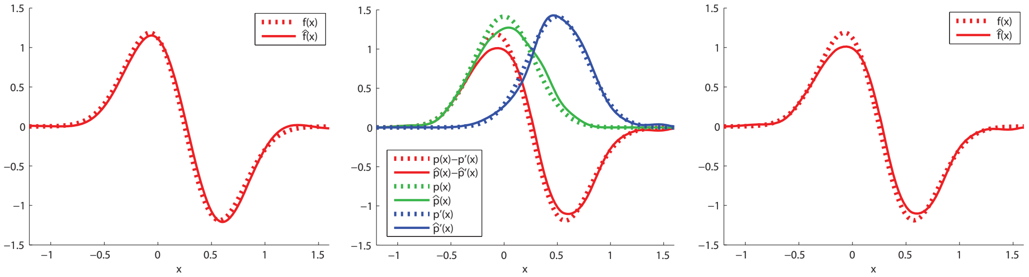
When $d = 5$ (Figure 3(b)), the KDE-based L^2-distance estimators even severely underestimate the true L^2-distance when μ is large. On the other hand, the LSDD-based L^2-distance estimator still gives reasonably accurate estimates of the true L^2-distance even when $d = 5$. However, we note that LSDD also slightly underestimates the true L^2-distance when μ is large, because slight underestimation tends to yield smaller variance and thus such stabilized solutions are more accurate in terms of the bias-variance trade-off.

Semi-Supervised Class-Balance Estimation: In real-world pattern recognition tasks, changes in class balance between the training and test phases are often observed. In such cases, naive classifier



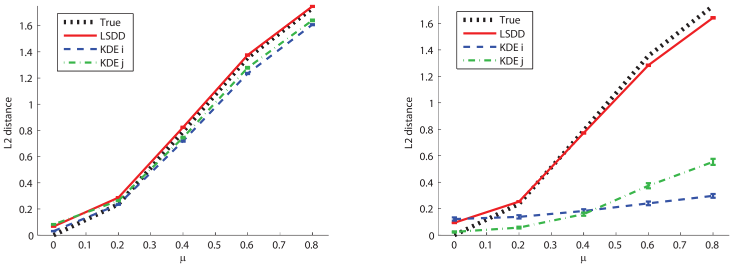
(a) LSDD (b) KDEi (c) KDEj

Figure 1: Estimation of density difference when $\mu = 0$ (i.e., $f(x) = p(x) - p'(x) = 0$).



(a) LSDD (b) KDEi (c) KDEj

Figure 2: Estimation of density difference when $\mu = 0.5$ (i.e., $f(x) = p(x) - p'(x) \neq 0$).



(a) $d = 1$ (b) $d = 5$

Figure 3: L^2-distance estimation by LSDD, KDEi, and KDEj for $n = n' = 200$ as functions of the Gaussian mean μ. Means and standard errors over 1000 runs are plotted.

training produces significant estimation bias because the class balance in the training dataset does not properly reflect that of the test dataset.

Here, we consider a binary pattern recognition task of classifying pattern $\boldsymbol{x} \in \mathbb{R}^d$ to class $y \in \{+1, -1\}$. Our goal is to learn the class balance of a test dataset in a semi-supervised learning setup where unlabeled test samples are provided in addition to labeled training samples [16]. The class balance in the test set can be estimated by matching a mixture of class-wise training input densities,

$$q_{\mathrm{test}}(\boldsymbol{x}; \pi) := \pi p_{\mathrm{train}}(\boldsymbol{x}|y = +1) + (1 - \pi) p_{\mathrm{train}}(\boldsymbol{x}|y = -1),$$

to the test input density $p_{\mathrm{test}}(\boldsymbol{x})$ [5], where $\pi \in [0, 1]$ is a mixing coefficient to learn. See Figure 4 for schematic illustration. Here, we use the L^2-distance estimated by LSDD and the difference of KDEs for this distribution matching. Note that, when LSDD is used to estimate the L^2-distance, separate estimation of $p_{\mathrm{train}}(\boldsymbol{x}|y = \pm 1)$ is not involved, but the difference between $p_{\mathrm{test}}(\boldsymbol{x})$ and $q_{\mathrm{test}}(\boldsymbol{x}; \pi)$ is directly estimated.

We use four UCI benchmark datasets (`http://archive.ics.uci.edu/ml/`), where we randomly choose 10 labeled training samples from each class and 50 unlabeled test samples following true class-prior $\pi^* = 0.1, 0.2, \ldots, 0.9$. Figure 6 plots the mean and standard error of the squared difference between true and estimated class-balances π and the misclassification error by a weighted ℓ_2-regularized least-squares classifier [17] with weighted cross-validation [18] over 1000 runs. The results show that LSDD tends to provide better class-balance estimates than the KDEi-based, the KDEj-based, and the EM-based methods [5], which are translated into lower classification errors.

Unsupervised Change Detection: The objective of change detection is to discover abrupt property changes behind time-series data. Let $\boldsymbol{y}(t) \in \mathbb{R}^m$ be an m-dimensional time-series sample at time t, and let $\boldsymbol{Y}(t) := [\boldsymbol{y}(t)^\top, \boldsymbol{y}(t+1)^\top, \ldots, \boldsymbol{y}(t+k-1)^\top]^\top \in \mathbb{R}^{km}$ be a subsequence of time series at time t with length k. We treat the subsequence $\boldsymbol{Y}(t)$ as a sample, instead of a single point $\boldsymbol{y}(t)$, by which time-dependent information can be incorporated naturally [6]. Let $\mathcal{Y}(t)$ be a set of r retrospective subsequence samples starting at time t: $\mathcal{Y}(t) := \{\boldsymbol{Y}(t), \boldsymbol{Y}(t+1), \ldots, \boldsymbol{Y}(t+r-1)\}$. Our strategy is to compute a certain dissimilarity measure between two consecutive segments $\mathcal{Y}(t)$ and $\mathcal{Y}(t+r)$, and use it as the plausibility of change points (see Figure 5). As a dissimilarity measure, we use the L^2-distance estimated by LSDD and the Kullback-Leibler (KL) divergence estimated by the *KL importance estimation procedure* (KLIEP) [2, 3]. We set $k = 10$ and $r = 50$.

First, we use the *IPSJ SIG-SLP Corpora and Environments for Noisy Speech Recognition* (CENSREC) dataset (`http://research.nii.ac.jp/src/en/CENSREC-1-C.html`). This dataset is provided by the *National Institute of Informatics, Japan* that records human voice in a noisy environment such as a restaurant. The top graphs in Figure 7(a) display the original time-series (true change points were manually annotated) and change scores obtained by KLIEP and LSDD. The graphs show that the LSDD-based change score indicates the existence of change points more clearly than the KLIEP-based change score.

Next, we use a dataset taken from the *Human Activity Sensing Consortium (HASC) challenge 2011* (`http://hasc.jp/hc2011/`), which provides human activity information collected by portable three-axis accelerometers. Because the orientation of the accelerometers is not necessarily fixed, we take the ℓ_2-norm of the 3-dimensional data. The HASC dataset is relatively simple, so we artificially added zero-mean Gaussian noise with standard deviation 5 at each time point with probability 0.005. The top graphs in Figure 7(b) display the original time-series for a sequence of actions “jog”, “stay”, “stair down”, “stay”, and “stair up” (there exists 4 change points at time 540, 1110, 1728, and 2286) and the change scores obtained by KLIEP and LSDD. The graphs show that the LSDD score is much more stable and interpretable than the KLIEP score.

Finally, we compare the change-detection performance more systematically using the *receiver operating characteristic (ROC) curves* (i.e., the false positive rate vs. the true positive rate) and the *area under the ROC curve (AUC) values*. In addition to LSDD and KLIEP, we test the L^2-distance estimated by KDEi and KDEj and native change detection methods based on autoregressive models (AR) [19], subspace identification (SI) [20], singular spectrum transformation (SST) [21], one-class support vector machine (SVM) [22], kernel Fisher discriminant analysis (KFD) [23], and kernel change-point detection (KCP) [24]. Tuning parameters included in these methods were manually optimized. For 10 datasets taken from each of the CENSREC and HASC data collections, mean ROC curves and AUC values are displayed at the bottom of Figure 7(b). The results show that LSDD tends to outperform other methods and is comparable to state-of-the-art native change-detection methods.

5 Conclusions

In this paper, we proposed a method for directly estimating the difference between two probability density functions without density estimation. The proposed method, called the *least-squares density-difference* (LSDD), was derived within the framework of kernel least-squares estimation, and its solution can be computed analytically in a computationally efficient and stable manner. Furthermore, LSDD is equipped with cross-validation, and thus all tuning parameters such as the kernel width and the regularization parameter can be systematically and objectively optimized. We derived a finite-sample error bound for LSDD in a non-parametric setup, and showed that it achieves the optimal convergence rate. We also proposed an L^2-distance estimator based on LSDD, which nicely cancels a bias caused by regularization. Through experiments on class-prior estimation and change-point detection, the usefulness of the proposed LSDD was demonstrated.

Acknowledgments: We would like to thank Wittawat Jitkrittum for his comments and Zaïd Harchaoui for providing us a program code of kernel change-point detection. MS was supported by MEXT KAKENHI 23300069 and AOARD, TK was supported by MEXT KAKENHI 24500340, TS was supported by MEXT KAKENHI 22700289, the Aihara Project, the FIRST program from JSPS initiated by CSTP, and the Global COE Program “The research and training center for new development in mathematics”, MEXT, Japan, MCdP was supported by MEXT Scholarship, SL was supported by the JST PRESTO program, and IT was supported by MEXT KAKENHI 23700165.

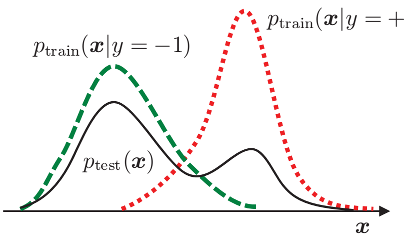


Figure 4: Class-balance estimation.

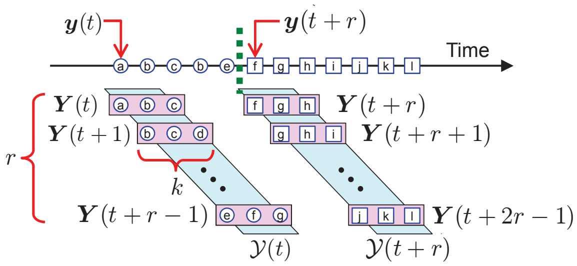
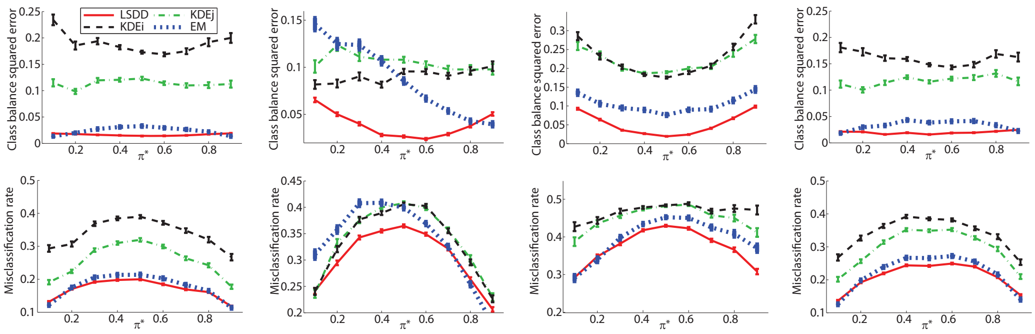
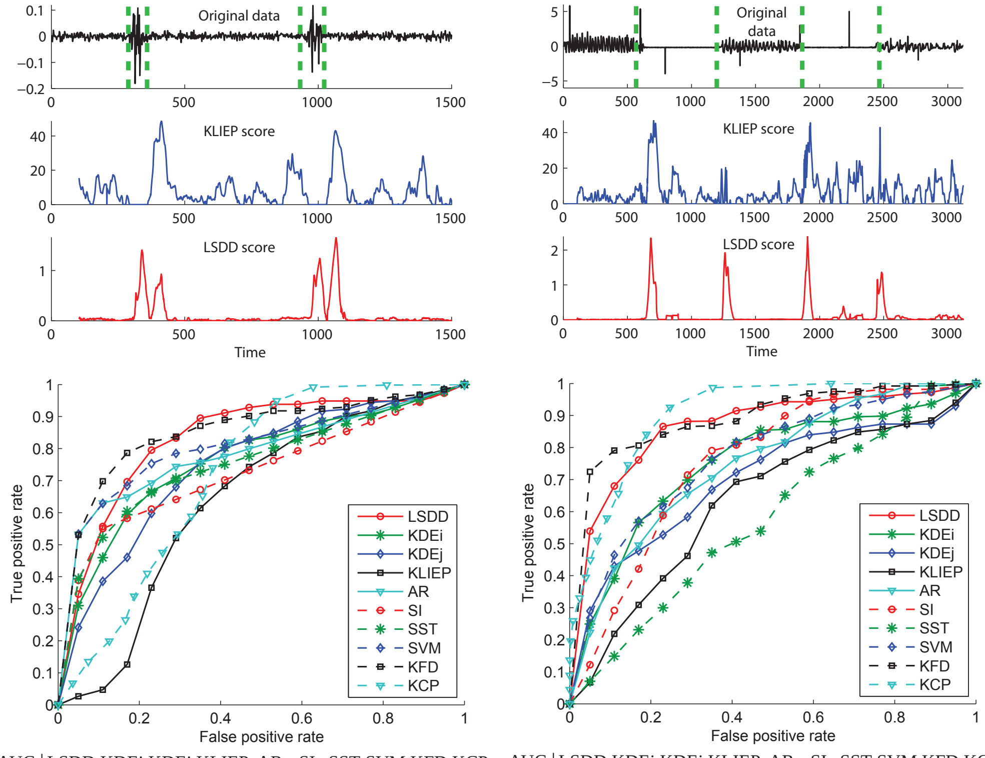


Figure 5: Change-point detection.



(a) Australian dataset (b) Diabetes dataset (c) German dataset (d) Statlogheart dataset

Figure 6: Results of semi-supervised class-balance estimation. Top: Squared error of class balance estimation. Bottom: Misclassification error by a weighted ℓ_2-regularized least-squares classifier.



AUC	LSDD	KDEi	KDEj	KLIEP	AR	SI	SST	SVM	KFD	KCP
Mean	**.879**	.755	.705	.635	.749	.756	.580	.773	**.905**	**.913**
SE	.014	.016	.023	.030	.013	.012	.023	.032	.013	.024

(a) Speech data

AUC	LSDD	KDEi	KDEj	KLIEP	AR	SI	SST	SVM	KFD	KCP
Mean	**.843**	.764	.751	.638	**.799**	.762	.764	**.815**	**.856**	.730
SE	.013	.029	.036	.020	.026	.020	.016	.018	.023	.032

(b) Accelerometer data

Figure 7: Results of unsupervised change detection. From top to bottom: Original time-series, change scores obtained by KLIEP and LSDD, mean ROC curves over 10 datasets, and AUC values for 10 datasets. The best method and comparable ones in terms of mean AUC values by the *t-test* at the significance level 5% are indicated with boldface. “SE” stands for “Standard error”.

References

[1] V. N. Vapnik. *Statistical Learning Theory*. Wiley, New York, NY, USA, 1998.

[2] M. Sugiyama, T. Suzuki, S. Nakajima, H. Kashima, P. von Bünau, and M. Kawanabe. Direct importance estimation for covariate shift adaptation. *Annals of the Institute of Statistical Mathematics*, 60(4):699–746, 2008.

[3] X. Nguyen, M. J. Wainwright, and M. I. Jordan. Estimating divergence functionals and the likelihood ratio by convex risk minimization. *IEEE Transactions on Information Theory*, 56(11):5847–5861, 2010.

[4] C. Cortes, Y. Mansour, and M. Mohri. Learning bounds for importance weighting. In *Advances in Neural Information Processing Systems 23*, pages 442–450, 2010.

[5] M. Saerens, P. Latinne, and C. Decaestecker. Adjusting the outputs of a classifier to new a priori probabilities: A simple procedure. *Neural Computation*, 14(1):21–41, 2002.

[6] Y. Kawahara and M. Sugiyama. Sequential change-point detection based on direct density-ratio estimation. *Statistical Analysis and Data Mining*, 5(2):114–127, 2012.

[7] N. Anderson, P. Hall, and D. Titterington. Two-sample test statistics for measuring discrepancies between two multivariate probability density functions using kernel-based density estimates. *Journal of Multivariate Analysis*, 50(1):41–54, 1994.

[8] A. Basu, I. R. Harris, N. L. Hjort, and M. C. Jones. Robust and efficient estimation by minimising a density power divergence. *Biometrika*, 85(3):549–559, 1998.

[9] P. Hall and M. P. Wand. On nonparametric discrimination using density differences. *Biometrika*, 75(3):541–547, 1988.

[10] M. Eberts and I. Steinwart. Optimal learning rates for least squares SVMs using Gaussian kernels. In *Advances in Neural Information Processing Systems 24*, pages 1539–1547, 2011.

[11] R. H. Farrell. On the best obtainable asymptotic rates of convergence in estimation of a density function at a point. *The Annals of Mathematical Statistics*, 43(1):170–180, 1972.

[12] B. W. Silverman. *Density Estimation for Statistics and Data Analysis*. Chapman and Hall, London, UK, 1986.

[13] E. Parzen. On the estimation of a probability density function and mode. *The Annals of Mathematical Statistics*, 33(3):1065–1076, 1962.

[14] R. T. Rockafellar. *Convex Analysis*. Princeton University Press, Princeton, NJ, USA, 1970.

[15] W. Härdle, M. Müller, S. Sperlich, and A. Werwatz. *Nonparametric and Semiparametric Models*. Springer, Berlin, Germany, 2004.

[16] O. Chapelle, B. Schölkopf, and A. Zien, editors. *Semi-Supervised Learning*. MIT Press, Cambridge, MA, USA, 2006.

[17] R. Rifkin, G. Yeo, and T. Poggio. Regularized least-squares classification. In *Advances in Learning Theory: Methods, Models and Applications*, pages 131–154. IOS Press, Amsterdam, the Netherlands, 2003.

[18] M. Sugiyama, M. Krauledat, and K.-R. Müller. Covariate shift adaptation by importance weighted cross validation. *Journal of Machine Learning Research*, 8:985–1005, May 2007.

[19] Y. Takeuchi and K. Yamanishi. A unifying framework for detecting outliers and change points from non-stationary time series data. *IEEE Transactions on Knowledge and Data Engineering*, 18(4):482–489, 2006.

[20] Y. Kawahara, T. Yairi, and K. Machida. Change-point detection in time-series data based on subspace identification. In *Proceedings of the 7th IEEE International Conference on Data Mining*, pages 559–564, 2007.

[21] V. Moskvina and A. A. Zhigljavsky. An algorithm based on singular spectrum analysis for change-point detection. *Communication in Statistics: Simulation & Computation*, 32(2):319–352, 2003.

[22] F. Desobry, M. Davy, and C. Doncarli. An online kernel change detection algorithm. *IEEE Transactions on Signal Processing*, 53(8):2961–2974, 2005.

[23] Z. Harchaoui, F. Bach, and E. Moulines. Kernel change-point analysis. In *Advances in Neural Information Processing Systems 21*, pages 609–616, 2009.

[24] S. Arlot, A. Celisse, and Z. Harchaoui. Kernel change-point detection. Technical Report 1202.3878, arXiv, 2012.

Least-Squares Two-Sample Test

Masashi Sugiyama
Tokyo Institute of Technology
and PRESTO, Japan Science and Technology Agency (JST),
2-12-1 O-okayama, Meguro-ku, Tokyo 152-8552, Japan.
sugi@cs.titech.ac.jp http://sugiyama-www.cs.titech.ac.jp/˜sugi

Taiji Suzuki
The University of Tokyo
7-3-1 Hongo, Bunkyo-ku, Tokyo 113-8656, Japan.
s-taiji@stat.t.u-tokyo.ac.jp

Yuta Itoh
Tokyo Institute of Technology
2-12-1 O-okayama, Meguro-ku, Tokyo 152-8552, Japan.
itoh@sg.cs.titech.ac.jp

Takafumi Kanamori
Nagoya University
Furocho, Chikusaku, Nagoya 464-8603, Japan.
kanamori@is.nagoya-u.ac.jp

Manabu Kimura
Tokyo Institute of Technology
2-12-1 O-okayama, Meguro-ku, Tokyo 152-8552, Japan.
kimura@sg.cs.titech.ac.jp

Abstract

The goal of the two-sample test (a.k.a. the homogeneity test) is, given two sets of samples, to judge whether the probability distributions behind the samples are the same or not. In this paper, we propose a novel non-parametric method of two-sample test based on a least-squares density ratio estimator. Through various experiments, we show that the proposed method overall produces smaller type-II error (i.e., the probability of judging the two distributions to be the same when they are actually different) than a state-of-the-art method, with slightly larger type-I error (i.e., the probability of judging the two distributions to be different when they are actually the same).

Keywords

two-sample test, homogeneity test, density ratio estimation, unconstrained least-squares importance fitting, Pearson divergence.

1 Introduction

Given two sets of samples, testing whether the probability distributions behind the samples are equivalent or not is a fundamental task in statistical data analysis. This problem is referred to as the *two-sample test* or the *homogeneity test* (Kullback, 1959).

1.1 Motivation of Two-Sample Test

The two-sample test is useful in various practically important learning scenarios. Here we describe some examples.

When learning is performed under non-stationary environment, e.g., in brain-computer interface (Sugiyama et al., 2007) and robot control (Hachiya et al., 2009), testing homogeneity of data generating distributions allows one to determine whether some adaptation scheme should be used or not. When the distributions are not significantly different, one can avoid using data-intensive non-stationarity adaptation techniques, which highly contributes to stabilizing the performance.

When multiple sets of data samples are available for learning, e.g., biological experimental results obtained from different laboratories (Borgwardt et al., 2006), the homogeneity test allows one to make a decision whether all the datasets are analyzed jointly as a single dataset or they should be treated separately.

Similarly, one can use the homogeneity test for deciding whether *multi-task learning* methods (Caruana et al., 1997) are employed or not. The rationale behind multi-task learning is that when several related learning tasks are provided, solving them simultaneously can give better solutions than solving them individually. However, when the tasks are not similar to each other, using multi-task learning techniques can degrade the performance. Thus, it is important to avoid using multi-task learning methods when the tasks are not similar to each other. This may be achieved by testing the homogeneity of datasets.

When several databases containing multiple fields are given, it is useful to identify the correspondence between fields by comparing underlying distributions since this allows one to merge databases (Gretton et al., 2007).

1.2 Methods of Two-Sample Test

The *t-test* (Student, 1908) is a classical method for testing homogeneity, which compares the means of two Gaussian distributions with common variance. Its multi-variate extension also exists (Hotelling, 1951). Although the t-test is a fundamental method for comparing the means, its range of application is limited to Gaussian distributions, which may not be fulfilled in practical applications.

The *Kolmogorov-Smirnov test* and the *Wald-Wolfowitz runs test* are classical non-parametric methods for the two-sample problem; their multi-dimensional variants have also been developed (Bickel, 1969; Friedman & Rafsky, 1979). Since then, different types of non-parametric tests have been studied (Anderson et al., 1994; Li, 1996).

Recently, a non-parametric extension of the t-test called the *maximum mean discrepancy* (MMD) was proposed (Borgwardt et al., 2006; Gretton et al., 2007). MMD compares the means of two distributions in a *universal reproducing kernel Hilbert space* (universal RKHS; Steinwart, 2001)—the Gaussian kernel is a typical example that induces a universal RKHS. MMD does not require a restrictive parametric assumption, so it could be a flexible alternative to the t-test. MMD was experimentally shown to outperform other homogeneity tests such as the *generalized Kolmogorov-Smirnov test* (Friedman & Rafsky, 1979), the *generalized Wald-Wolfowitz test* (Friedman & Rafsky, 1979), the *Hall-Tajvidi test* (Hall & Tajvidi, 2002), and the *Biau-Györfi test* (Biau & Györfi, 2005).

The performance of MMD depends on the choice of universal RKHSs (e.g., the Gaussian width in the case of Gaussian RKHSs). Thus, the universal RKHS should be carefully chosen for obtaining the state-of-the-art performance. The Gaussian RKHS with width set to the median distance between samples has been a popular heuristic in practice (Borgwardt et al., 2006; Gretton et al., 2007). Recently, a novel idea of using the universal RKHS (or the Gaussian widths) yielding the maximum MMD value has been introduced (Sriperumbudur et al., 2009).

1.3 Divergence Estimation

Another approach to the two-sample problem is to evaluate a divergence between two distributions. The divergence-based approach is advantageous in that cross-validation over the divergence functional is available for optimizing tuning parameters in a data-dependent manner. A typical choice of the divergence functional would be the *f-divergences* (Ali & Silvey, 1966; Csiszár, 1967), which includes the *Kullback-Leibler divergence* (Kullback & Leibler, 1951) and the *Pearson divergence* (Pearson, 1900) as special cases.

Various methods for estimating the divergence functional have been studied so far (Darbellay & Vajda, 1999; Wang et al., 2005; Silva & Narayanan, 2007; Pérez-Cruz, 2008). Among them, approaches based on *density ratio estimation* have been shown to be promising both theoretically and experimentally (Sugiyama et al., 2008; Gretton et al., 2009; Kanamori et al., 2009a; Nguyen et al., 2010). So far, a parametric density ratio estimator based on logistic regression (Qin, 1998; Cheng & Chu, 2004) has been applied to the test of homogeneity (Keziou & Leoni-Aubin, 2005).

Although the density ratio estimator based on logistic regression was proved to achieve the smallest asymptotic variance among a class of semi-parametric estimators (Qin, 1998), this theoretical guarantee is valid only when the parametric model is *correctly specified* (i.e., the target density ratio is included in the parametric model at hand). However, when this unrealistic assumption is violated, a divergence-based density ratio estimator (Sugiyama et al., 2008; Nguyen et al., 2010) was shown to perform better (Kanamori et al., 2010).

Among various divergence-based density ratio estimators, a method called *unconstrained least-squares importance fitting* (uLSIF) was demonstrated to be accurate and computationally efficient (Kanamori et al., 2009a). Furthermore, uLSIF was proved to possess the optimal non-parametric convergence rate and numerical stability (Kanamori et al., 2009b). In this paper, we therefore develop a new method for testing homogeneity based on uLSIF.

Similarly to MMD, our uLSIF-based homogeneity test processes data samples only through kernel functions. Thus, the proposed method can be used for testing the homogeneity of *non-vectorial structured objects* such as strings, trees, and graphs by employing kernel functions defined for such structured data (Lodhi et al., 2002; Duffy & Collins, 2002; Kashima & Koyanagi, 2002; Kondor & Lafferty, 2002; Kashima et al., 2003; Gärtner et al., 2003; Gärtner, 2003). This is an advantage over traditional two-sample tests.

1.4 Organization of This Paper

The rest of this paper is structured as follows. In Section 2, we review the uLSIF method for density ratio estimation. In Section 3, we describe a method of divergence estimation based on uLSIF, and investigate its theoretical properties. In Section 4, we give a two-sample test based on the permutation test (Efron & Tibshirani, 1993), which we call *least-squares two-sample test* (LSTT). We review the MMD method in Section 5, and compare the experimental performance of LSTT with MMD in Section 6. Finally, we conclude in Section 7.

2 Density Ratio Estimation

In this section, we consider the problem of density ratio estimation, and review a method called *unconstrained least-squares importance fitting* (uLSIF; Kanamori et al., 2009a), which will be used in the following sections. Since this section is devoted to reviewing uLSIF, those who are familiar with it may skip this section and directly go to the next section.

2.1 Formulation of Density Ratio Estimation

Suppose we are given a set of samples

$$\mathcal{X} := \{\boldsymbol{x}_i | \boldsymbol{x}_i \in \mathbb{R}^d\}_{i=1}^{n}$$

drawn independently from a probability distribution P with density $p(\boldsymbol{x})$, and another set of samples

$$\mathcal{X}' := \{\boldsymbol{x}'_j | \boldsymbol{x}'_j \in \mathbb{R}^d\}_{j=1}^{n'}$$

drawn independently from (possibly) another probability distribution P' with density $p'(\boldsymbol{x})$:

$$\{\boldsymbol{x}_i\}_{i=1}^{n} \overset{i.i.d.}{\sim} P,$$
$$\{\boldsymbol{x}'_j\}_{j=1}^{n'} \overset{i.i.d.}{\sim} P'.$$

The goal of density ratio estimation is to estimate the density ratio function

$$r(\boldsymbol{x}) := \frac{p(\boldsymbol{x})}{p'(\boldsymbol{x})} \tag{1}$$

from the samples $\mathcal{X}$ and $\mathcal{X}'$, where we assume $p'(\boldsymbol{x}) > 0$ for all $\boldsymbol{x}$.

2.2 Least-Squares Approach to Density Ratio Estimation

Let us model the density ratio function $r(\boldsymbol{x})$ by the following kernel model[1]:

$$\begin{aligned}\widehat{r}(\boldsymbol{x}) &:= \alpha_0 + \sum_{i=1}^{n} \alpha_i K(\boldsymbol{x}, \boldsymbol{x}_i)\\ &= \boldsymbol{\alpha}^\top \boldsymbol{k}(\boldsymbol{x}),\end{aligned}$$

where

$$\boldsymbol{\alpha} := (\alpha_0, \alpha_1, \ldots, \alpha_{n+1})^\top$$

are parameters to be learned from data samples, $^\top$ denotes the transpose of a matrix or a vector,

$$\boldsymbol{k}(\boldsymbol{x}) := (1, K(\boldsymbol{x}, \boldsymbol{x}_1), \ldots, K(\boldsymbol{x}, \boldsymbol{x}_n))^\top$$

are kernel basis functions. A popular choice of the kernel is the Gaussian function:

$$K(\boldsymbol{x}, \boldsymbol{x}') = \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{x}'\|^2}{2\sigma^2}\right), \tag{2}$$

where σ^2 denotes the Gaussian variance.

We determine the parameter $\boldsymbol{\alpha}$ in the model $\widehat{r}(\boldsymbol{x})$ so that the following squared-error J_0 is minimized:

$$\begin{aligned}J_0(\boldsymbol{\alpha}) &:= \frac{1}{2}\int \left(\widehat{r}(\boldsymbol{x}) - r(\boldsymbol{x})\right)^2 p'(\boldsymbol{x})\mathrm{d}\boldsymbol{x}\\ &= \frac{1}{2}\int \widehat{r}(\boldsymbol{x})^2 p'(\boldsymbol{x})\mathrm{d}\boldsymbol{x} - \int \widehat{r}(\boldsymbol{x}) p(\boldsymbol{x})\mathrm{d}\boldsymbol{x} + \frac{1}{2}\int r(\boldsymbol{x}) p(\boldsymbol{x})\mathrm{d}\boldsymbol{x},\end{aligned}$$

where the last term is a constant and therefore can be safely ignored. Let us denote the first two terms by J:

$$\begin{aligned}J(\boldsymbol{\alpha}) &:= \frac{1}{2}\int \widehat{r}(\boldsymbol{x})^2 p'(\boldsymbol{x})\mathrm{d}\boldsymbol{x} - \int \widehat{r}(\boldsymbol{x}) p(\boldsymbol{x})\mathrm{d}\boldsymbol{x}\\ &= \frac{1}{2}\boldsymbol{\alpha}^\top \boldsymbol{H}\boldsymbol{\alpha} - \boldsymbol{h}^\top \boldsymbol{\alpha},\end{aligned} \tag{3}$$

[1] We included the constant basis function, 1, in our model, which is different from the original uLSIF paper (Kanamori et al., 2009a). In the context of two-sample test, we empirically found that including the constant basis tends to improve the estimation accuracy since the density ratio function we approximate can be close to constant (i.e., $r(\boldsymbol{x}) \approx 1$) when the two distributions are similar.

where $\boldsymbol{H}$ is the $(n+1)\times(n+1)$ matrix defined by

$$\boldsymbol{H} := \int \boldsymbol{k}(\boldsymbol{x})\boldsymbol{k}(\boldsymbol{x})^\top p'(\boldsymbol{x})\mathrm{d}\boldsymbol{x},$$

and $\boldsymbol{h}$ is the $(n+1)$-dimensional vector defined by

$$\boldsymbol{h} := \int \boldsymbol{k}(\boldsymbol{x})p(\boldsymbol{x})\mathrm{d}\boldsymbol{x}.$$

2.3 Empirical Approximation

Since J contains the expectation over unknown densities $p(\boldsymbol{x})$ and $p'(\boldsymbol{x})$, we approximate the expectations by empirical averages. Then we obtain

$$\begin{aligned}\widehat{J}(\boldsymbol{\alpha}) &:= \frac{1}{2n'}\sum_{j=1}^{n'}\widehat{r}(\boldsymbol{x}'_j)^2 - \frac{1}{n}\sum_{i=1}^{n}\widehat{r}(\boldsymbol{x}_i)\\ &= \frac{1}{2}\boldsymbol{\alpha}^\top\widehat{\boldsymbol{H}}\boldsymbol{\alpha} - \boldsymbol{\alpha}^\top\widehat{\boldsymbol{h}},\end{aligned}$$

where $\widehat{\boldsymbol{H}}$ is the $(n+1)\times(n+1)$ matrix defined by

$$\widehat{\boldsymbol{H}} := \frac{1}{n'}\sum_{j=1}^{n'}\boldsymbol{k}(\boldsymbol{x}'_j)\boldsymbol{k}(\boldsymbol{x}'_j)^\top,$$

and $\widehat{\boldsymbol{h}}$ is the $(n+1)$-dimensional vector defined by

$$\widehat{\boldsymbol{h}} := \frac{1}{n}\sum_{i=1}^{n}\boldsymbol{k}(\boldsymbol{x}_i). \tag{4}$$

By including a regularization term, the uLSIF optimization problem is formulated as follows.

$$\widehat{\boldsymbol{\alpha}} := \underset{\boldsymbol{\alpha}}{\mathrm{argmin}}\left[\frac{1}{2}\boldsymbol{\alpha}^\top\widehat{\boldsymbol{H}}\boldsymbol{\alpha} - \boldsymbol{\alpha}^\top\widehat{\boldsymbol{h}} + \frac{\lambda}{2}\boldsymbol{\alpha}^\top\boldsymbol{\alpha}\right], \tag{5}$$

where $\boldsymbol{\alpha}^\top\boldsymbol{\alpha}/2$ is a regularizer and λ (≥ 0) is the regularization parameter that controls the strength of regularization. By taking the derivative of the above objective function with respect to the parameter $\boldsymbol{\alpha}$ and equating it to zero, we can analytically obtain the solution $\widehat{\boldsymbol{\alpha}}$ as

$$\widehat{\boldsymbol{\alpha}} = (\widehat{\boldsymbol{H}} + \lambda\boldsymbol{I}_{n+1})^{-1}\widehat{\boldsymbol{h}}, \tag{6}$$

where $\boldsymbol{I}_{n+1}$ is the $(n+1)$-dimensional identity matrix. Finally, the density ratio estimator $\widehat{r}(\boldsymbol{x})$ is given by

$$\widehat{r}(\boldsymbol{x}) := \widehat{\boldsymbol{\alpha}}^\top\boldsymbol{k}(\boldsymbol{x}).$$

Thanks to the analytic-form expression, uLSIF is computationally more efficient than alternative density ratio estimators which involve non-linear optimization (Qin, 1998; Cheng & Chu, 2004; Huang et al., 2007; Sugiyama et al., 2008; Nguyen et al., 2010). It was theoretically shown that uLSIF possesses the optimal non-parametric convergence rate and optimal numerical stability (Kanamori et al., 2009b).

2.4 Model Selection by Cross-Validation

The practical performance of uLSIF depends on the choice of the kernel function (the kernel width σ in the case of Gaussian kernel (2)) and the regularization parameter λ. Model selection of uLSIF is possible based on *cross-validation* with respect to the error criterion J defined by Eq.(3) (Kanamori et al., 2009a).

More specifically, each of the sample sets $\mathcal{X} = \{\boldsymbol{x}_i\}_{i=1}^{n}$ and $\mathcal{X}' = \{\boldsymbol{x}'_j\}_{j=1}^{n'}$ is divided into M disjoint sets[2] $\{\mathcal{X}_m\}_{m=1}^{M}$ and $\{\mathcal{X}'_m\}_{m=1}^{M}$. Then an uLSIF solution $\widehat{r}_m(\boldsymbol{x})$ is obtained using $\mathcal{X}\backslash\mathcal{X}_m$ and $\mathcal{X}'\backslash\mathcal{X}'_m$ (i.e., all samples without $\mathcal{X}_m$ and $\mathcal{X}'_m$), and its J-value for the hold-out samples $\mathcal{X}_m$ and $\mathcal{X}'_m$ is computed as

$$\widehat{J}_m^{\mathrm{CV}} := \frac{1}{2|\mathcal{X}'_m|}\sum_{\boldsymbol{x}'\in\mathcal{X}'_m}\widehat{r}_m(\boldsymbol{x}')^2 - \frac{1}{|\mathcal{X}_m|}\sum_{\boldsymbol{x}\in\mathcal{X}_m}\widehat{r}_m(\boldsymbol{x}),$$

where $|\mathcal{X}|$ denotes the number of elements in the set $\mathcal{X}$. This procedure is repeated for $m = 1, \ldots, M$, and the average of $\widehat{J}_m^{\mathrm{CV}}$ over all m is computed as

$$\widehat{J}^{\mathrm{CV}} := \frac{1}{M}\sum_{m=1}^{M}\widehat{J}_m^{\mathrm{CV}}.$$

Finally, the model (the kernel width σ and the regularization parameter λ in the current setup) that minimizes $\widehat{J}^{\mathrm{CV}}$ is chosen as the most suitable one.

3 Divergence Estimation

In this section, we describe a divergence estimator based on uLSIF, and investigate its theoretical properties.

[2] $M = 5$ seems to be a popular choice (Hastie et al., 2001). We also follow this ‘rule-of-thumb’ choice in this paper.

3.1 Formulation of Divergence Estimation

Let us consider the *Pearson divergence* (Pearson, 1900) from P to P' as a discrepancy measure between P and P', which is defined and expressed as follows:

$$\begin{aligned}\mathrm{PE}(P, P') &:= \frac{1}{2}\int \left(\frac{p(\boldsymbol{x})}{p'(\boldsymbol{x})} - 1\right)^2 p'(\boldsymbol{x})\mathrm{d}\boldsymbol{x} \\ &= \frac{1}{2}\int r(\boldsymbol{x})p(\boldsymbol{x})\mathrm{d}\boldsymbol{x} - \int r(\boldsymbol{x})p'(\boldsymbol{x})\mathrm{d}\boldsymbol{x} + \frac{1}{2},\end{aligned} \tag{7}$$

where $r(\boldsymbol{x})$ is the density ratio function defined by

$$r(\boldsymbol{x}) = \frac{p(\boldsymbol{x})}{p'(\boldsymbol{x})}.$$

$\mathrm{PE}(P, P')$ vanishes if and only if $P = P'$. The Pearson divergence is a squared-loss variant of the *Kullback-Leibler divergence* (Kullback & Leibler, 1951), and is an instance of the *f-divergences*, which are also known as the *Csiszár f-divergences* (Csiszár, 1967) or the *Ali-Silvey distances* (Ali & Silvey, 1966).

3.2 uLSIF-based Pearson Divergence Estimation

Approximating the expectations in Eq.(7) by empirical averages and replacing the density ratio function $r(\boldsymbol{x})$ by an uLSIF-based estimator $\widehat{r}(\boldsymbol{x})$, we have the following Pearson divergence estimator:

$$\begin{aligned}\widehat{\mathrm{PE}}(\mathcal{X}, \mathcal{X}') &:= \frac{1}{2n}\sum_{i=1}^{n}\widehat{r}(\boldsymbol{x}_i) - \frac{1}{n'}\sum_{j=1}^{n'}\widehat{r}(\boldsymbol{x}'_j) + \frac{1}{2} \\ &= \frac{1}{2}\widehat{\boldsymbol{\alpha}}^\top\widehat{\boldsymbol{h}} - \widehat{\boldsymbol{\alpha}}^\top\widehat{\boldsymbol{h}}' + \frac{1}{2},\end{aligned} \tag{8}$$

where $\widehat{\boldsymbol{\alpha}}$ is given by Eq.(6), $\widehat{\boldsymbol{h}}$ is defined by Eq.(4), and $\widehat{\boldsymbol{h}}'$ is the $(n+1)$-dimensional vector defined by

$$\widehat{\boldsymbol{h}}' := \frac{1}{n'}\sum_{j=1}^{n'}\boldsymbol{k}(\boldsymbol{x}'_j).$$

Note that $\widehat{\mathrm{PE}}(\mathcal{X}, \mathcal{X}')$ can take a negative value, although the true $\mathrm{PE}(P, P')$ is non-negative by definition. Thus, the estimation accuracy of $\widehat{\mathrm{PE}}(\mathcal{X}, \mathcal{X}')$ can be improved by taking its positive part by rounding up a negative estimate to zero. However, we do not employ this rounding-up strategy here since we are interested in the relative *ranking* of the divergence estimates, as explained in Section 4.1.

3.3 Theoretical Properties

Here, let us theoretically investigate asymptotic properties of the uLSIF-based Pearson divergence estimator $\widehat{\mathrm{PE}}(\mathcal{X},\mathcal{X}')$. More specifically, we show the asymptotic convergence rate of our non-parametric estimator $\widehat{\mathrm{PE}}(\mathcal{X},\mathcal{X}')$ to the true $\mathrm{PE}(P,P')$.

Since the derivation of the convergence rate is highly technical, we defer all the technical details in Appendix A. Here, we focus on explaining the insight we can gain from our theoretical analysis.

Theorem 1. *Under the technical assumptions described in Appendix A, we have*

$$|\widehat{\mathrm{PE}}(\mathcal{X},\mathcal{X}') - \mathrm{PE}(P,P')| = \mathcal{O}_p\left(\left(\frac{\log \overline{n}}{\overline{n}}\right)^{\frac{2}{2+\gamma}} + C\left(\frac{\log \overline{n}}{\overline{n}}\right)^{\frac{1}{2+\gamma}}\right), \tag{9}$$

where

$$C := \sqrt{\int (r(\boldsymbol{x})-1)^2 p'(\boldsymbol{x})\mathrm{d}\boldsymbol{x}}. \tag{10}$$

$\mathcal{O}_p$ denotes the asymptotic order in probability, $\overline{n} := \min(n,n')$, and γ $(0<\gamma<1)$ is a constant determined by the kernel function $K(\cdot,\cdot)$.

The above theorem means that the convergence rate of $\widehat{\mathrm{PE}}(\mathcal{X},\mathcal{X}')$ to $\mathrm{PE}(P,P')$ is $\left(\frac{\log \overline{n}}{\overline{n}}\right)^{\frac{1}{2+\gamma}}$ in general. However, when the two distributions P and P' are the same, $r(\boldsymbol{x})=1$ and thus $C=0$ (see Eq.(10)). Then, the $\mathcal{O}_p\left(\left(\frac{\log \overline{n}}{\overline{n}}\right)^{\frac{1}{2+\gamma}}\right)$-term in Eq.(9) disappears, and therefore our estimator possesses an even faster convergence rate $\mathcal{O}_p\left(\left(\frac{\log \overline{n}}{\overline{n}}\right)^{\frac{2}{2+\gamma}}\right)$.

4 Least-Squares Two-Sample Test

Theoretical properties of our Pearson divergence estimator $\widehat{\mathrm{PE}}(\mathcal{X},\mathcal{X}')$ have been elucidated above. In this section, we propose a two-sample test based on $\widehat{\mathrm{PE}}(\mathcal{X},\mathcal{X}')$. We first describe a basic procedure of our two-sample test, and study its theoretical properties. Then we illustrate its behavior using toy datasets, and discuss practical issues for improving the performance.

4.1 Permutation Test with Finite Samples

Our two-sample test is based on the *permutation test* (Efron & Tibshirani, 1993).

We first run the uLSIF-based Pearson divergence estimation procedure using the original datasets $\mathcal{X}$ and $\mathcal{X}'$, and obtain a Pearson divergence estimate $\widehat{\mathrm{PE}}(\mathcal{X},\mathcal{X}')$. Next, we randomly permute the $|\mathcal{X}\cup\mathcal{X}'|$ samples, and assign the first $|\mathcal{X}|$ samples to a set $\widetilde{\mathcal{X}}$ and the remaining $|\mathcal{X}'|$ samples to another set $\widetilde{\mathcal{X}}'$. Then we run the uLSIF-based Pearson

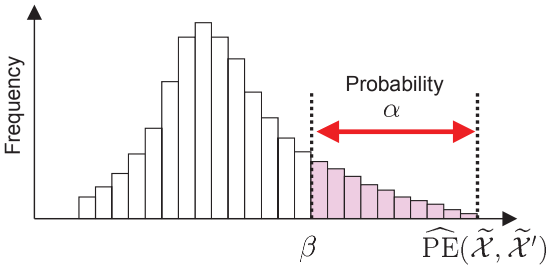


Figure 1: The role of the variables α and β in Theorem 2.

divergence estimation procedure again using the randomly shuffled datasets $\widetilde{\mathcal{X}}$ and $\widetilde{\mathcal{X}}'$, and obtain a Pearson divergence estimate $\widehat{\mathrm{PE}}(\widetilde{\mathcal{X}}, \widetilde{\mathcal{X}}')$. Since $\widetilde{\mathcal{X}}$ and $\widetilde{\mathcal{X}}'$ can be regarded as being drawn from the same distribution, $\widehat{\mathrm{PE}}(\widetilde{\mathcal{X}}, \widetilde{\mathcal{X}}')$ would take a value close to zero. This random shuffling procedure is repeated many times, and the distribution of $\widehat{\mathrm{PE}}(\widetilde{\mathcal{X}}, \widetilde{\mathcal{X}}')$ under the null-hypothesis (i.e., the two distributions are the same) is constructed. Finally, the p-value is approximated by evaluating the relative ranking of $\widehat{\mathrm{PE}}(\mathcal{X}, \mathcal{X}')$ in the distribution of $\widehat{\mathrm{PE}}(\widetilde{\mathcal{X}}, \widetilde{\mathcal{X}}')$.

We refer to this procedure as the *least-squares two-sample test* (LSTT).

4.2 Theoretical Properties

Here, we investigate theoretical properties of the above permutation procedure under the null-hypothesis $P = P'$.

Theorem 2. *Suppose $|\mathcal{X}| = |\mathcal{X}'|$, and let F be the distribution function of $\widehat{\mathrm{PE}}(\widetilde{\mathcal{X}}, \widetilde{\mathcal{X}}')$. Let*

$$\beta := \sup\{t \in \mathbb{R} \mid F(t) \leq 1 - \alpha\}$$

be the upper 100α-percentile point of F (see Figure 1). If $P = P'$, we have

$$\mathrm{Prob}\left(\widehat{\mathrm{PE}}(\mathcal{X}, \mathcal{X}') > \beta\right) \leq \alpha,$$

where '$\mathrm{Prob}(e)$' denotes the probability of an event e.

A proof of Theorem 2 is provided in Appendix B.

Theorem 2 means that, for a given significance level[3] α, the probability that $\widehat{\mathrm{PE}}(\mathcal{X}, \mathcal{X}')$ exceeds β is at most α when $P = P'$. Thus, when the null hypothesis is correct, it will be properly accepted with a specified probability.

[3] Conventionally, $\alpha = 0.01$ or 0.05 is used.

4.3 Numerical Examples

Let the number of samples be $n = n' = 500$, and

$$\mathcal{X} = \{\boldsymbol{x}_i\}_{i=1}^{n} \overset{i.i.d.}{\sim} P = N(0,1),$$
$$\mathcal{X}' = \{\boldsymbol{x}'_j\}_{j=1}^{n'} \overset{i.i.d.}{\sim} P' = N(\mu,\sigma^2),$$

where $N(\mu,\sigma^2)$ denotes the normal distribution with mean μ and variance σ^2. We consider the following four setups:

(a) $(\mu,\sigma) = (0, 1.3)$: P' has larger standard deviation than P,

(b) $(\mu,\sigma) = (0, 0.7)$: P' has smaller standard deviation than P,

(c) $(\mu,\sigma) = (0.3, 1)$: P and P' have different means,

(d) $(\mu,\sigma) = (0, 1)$: P and P' are the same.

Histograms of $\mathcal{X} = \{\boldsymbol{x}_i\}_{i=1}^{n}$ and $\mathcal{X}' = \{\boldsymbol{x}'_j\}_{j=1}^{n'}$ for the above four cases are depicted in Figure 2. Examples of randomly shuffled samples $\widetilde{\mathcal{X}}$ are also plotted at the bottom, where $\widetilde{\mathcal{X}}$ is thought to follow $\frac{1}{2}N(0,1) + \frac{1}{2}N(\mu,\sigma^2)$. Since $\widetilde{\mathcal{X}}'$ has a similar histogram to $\widetilde{\mathcal{X}}$, its plot is omitted.

Figure 3 depicts histograms of $\widehat{\mathrm{PE}}(\widetilde{\mathcal{X}}, \widetilde{\mathcal{X}}')$ (i.e., shuffled datasets), showing that the profiles of the null distribution (i.e., the two distributions are the same) are rather similar to each other for the four cases. The values of $\widehat{\mathrm{PE}}(\mathcal{X}, \mathcal{X}')$ (i.e., the original datasets) are also plotted in Figure 3 using the '×'-symbol on the horizontal axis, showing that the p-values tends to be small when $P \neq P'$ and the p-value is large when $P = P'$. This is desirable behavior as a hypothesis test.

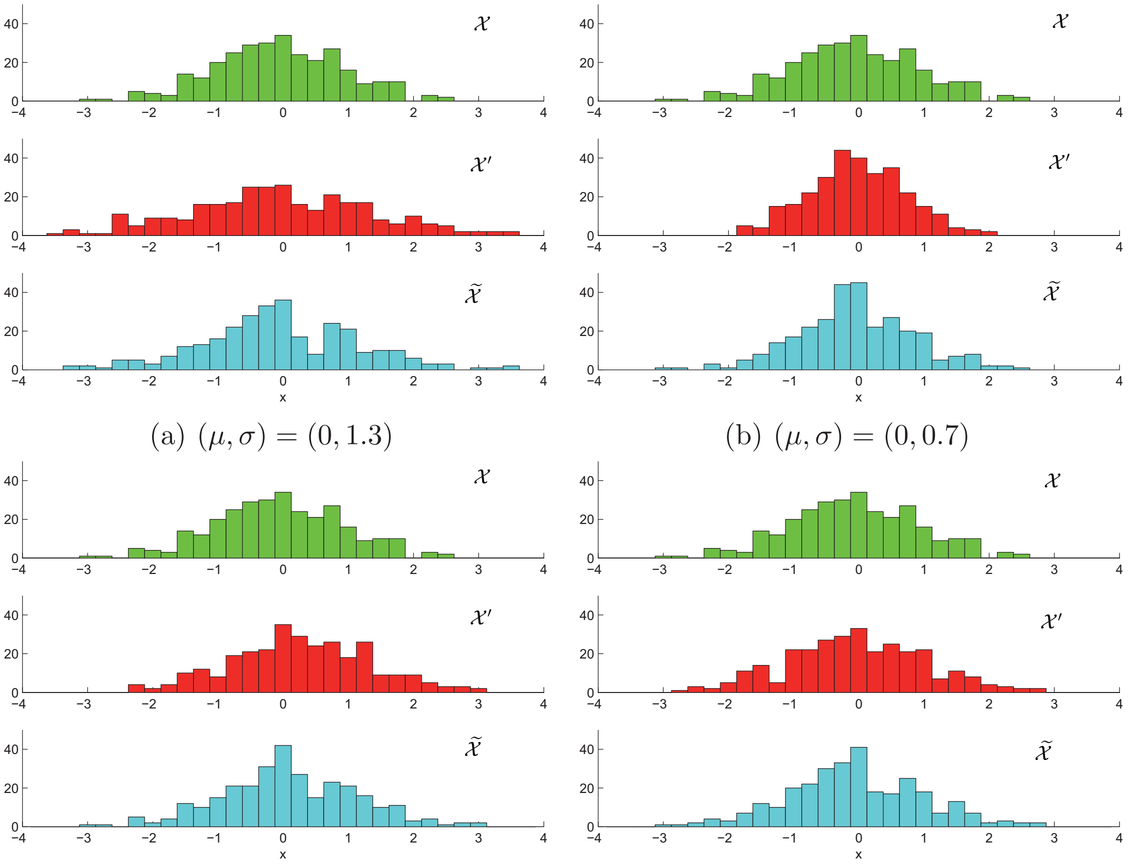
Figure 4 depicts the mean and standard deviation of p-values over 100 runs as functions of the sample size n $(= n')$, indicated by 'plain'. The graphs show that, when $P \neq P'$, the p-values tend to decrease as n increases. On the other hand, when $P = P'$, the p-values are almost unchanged and kept to relatively large values.

Figure 5 depicts the rate of accepting the null hypothesis (i.e., $P = P'$) over 100 runs when the significance level is set to 0.05 (i.e., the rate of p-values larger than 0.05). The graphs show that, when $P \neq P'$, the null hypothesis tends to be more frequently rejected as n increases. On the other hand, when $P = P'$, the null hypothesis is almost always accepted. Thus, the proposed test was shown to work properly for these toy datasets.

4.4 Choice of Numerator/Denominator Densities

In our test procedure, we are using uLSIF for estimating the density ratio function $r(\boldsymbol{x})$:

$$r(\boldsymbol{x}) = \frac{p(\boldsymbol{x})}{p'(\boldsymbol{x})}.$$



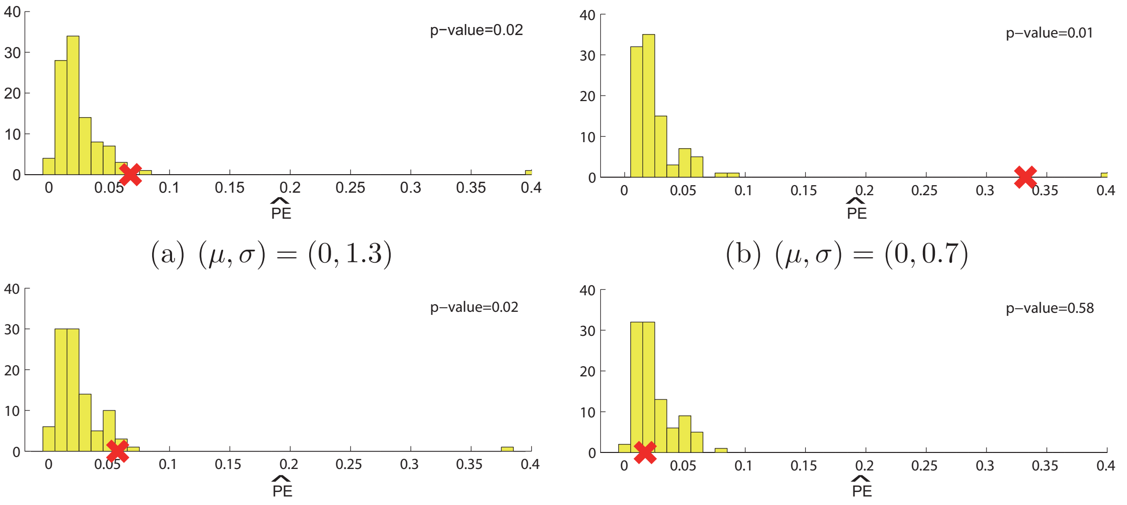
(a) $(\mu, \sigma) = (0, 1.3)$

(b) $(\mu, \sigma) = (0, 0.7)$

(c) $(\mu, \sigma) = (0.3, 1)$

(d) $(\mu, \sigma) = (0, 1)$

Figure 2: Histograms of original samples $\mathcal{X} \sim N(0,1)$ and $\mathcal{X}' \sim N(\mu, \sigma^2)$, and the shuffled samples (which are thought to follow $\widetilde{\mathcal{X}} \sim \frac{1}{2}N(0,1) + \frac{1}{2}N(\mu, \sigma^2)$) for the toy datasets.



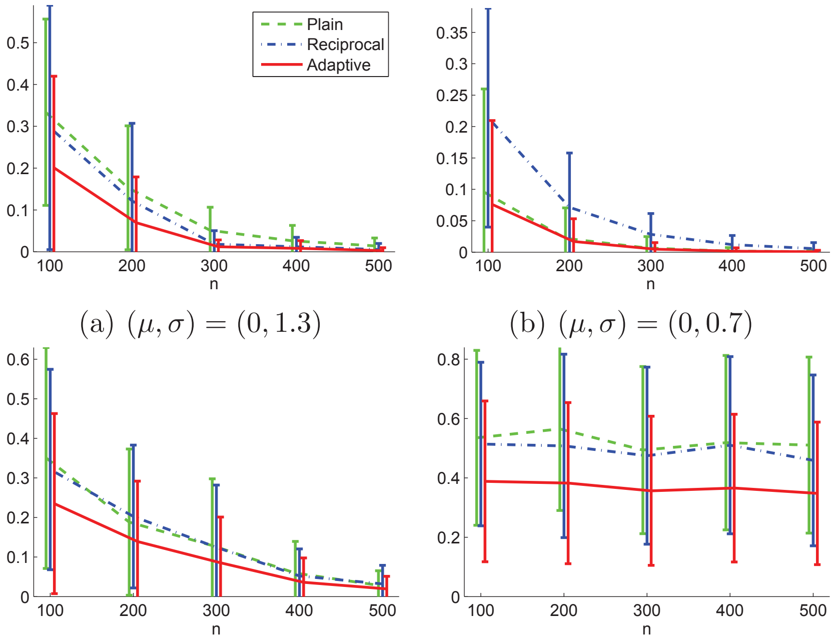
(a) $(\mu, \sigma) = (0, 1.3)$

(b) $(\mu, \sigma) = (0, 0.7)$

(c) $(\mu, \sigma) = (0.3, 1)$

(d) $(\mu, \sigma) = (0, 1)$

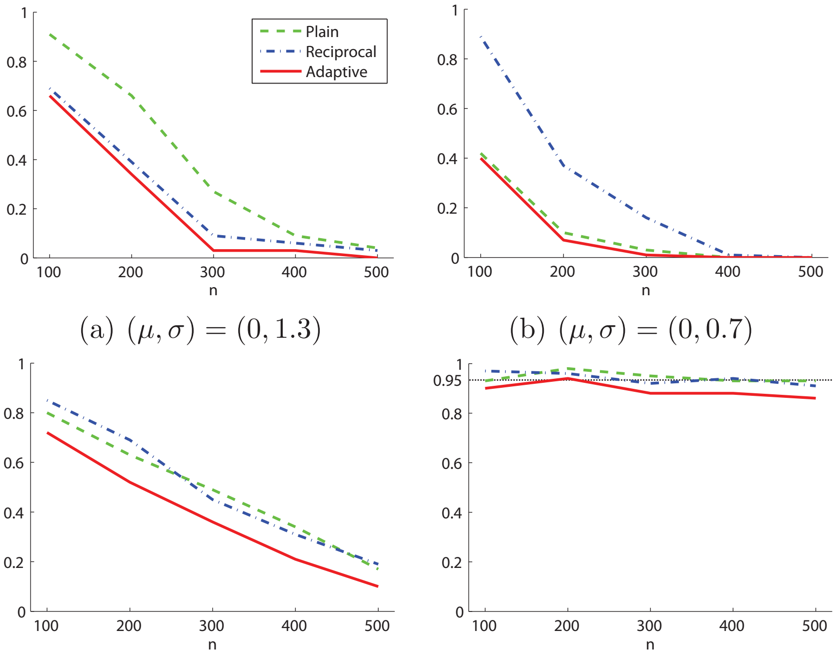
Figure 3: Histograms of $\widehat{\text{PE}}(\widetilde{\mathcal{X}}, \widetilde{\mathcal{X}}')$ (i.e., shuffled datasets) for the toy datasets. '×' indicates the value of $\widehat{\text{PE}}(\mathcal{X}, \mathcal{X}')$ (i.e., the original datasets).



(a) $(\mu, \sigma) = (0, 1.3)$ (b) $(\mu, \sigma) = (0, 0.7)$

(c) $(\mu, \sigma) = (0.3, 1)$ (d) $(\mu, \sigma) = (0, 1)$

Figure 4: Mean and standard deviation of p-values for the toy datasets.



(a) $(\mu, \sigma) = (0, 1.3)$ (b) $(\mu, \sigma) = (0, 0.7)$

(c) $(\mu, \sigma) = (0.3, 1)$ (d) $(\mu, \sigma) = (0, 1)$

Figure 5: The rate of accepting the null hypothesis (i.e., $P = P'$) for the toy datasets under the significance level 0.05.

By definition, the *reciprocal* of the density ratio $r(\boldsymbol{x})$,

$$\frac{1}{r(\boldsymbol{x})} = \frac{p'(\boldsymbol{x})}{p(\boldsymbol{x})},$$

is also a density ratio function, assuming that $p(\boldsymbol{x}) > 0$ for all $\boldsymbol{x}$. This means that we can use uLSIF in two ways, either estimating the original density ratio $r(\boldsymbol{x})$ or its reciprocal $1/r(\boldsymbol{x})$.

To illustrate this difference, we also carried out the same experiments as Section 4.3 by swapping $\mathcal{X}$ and $\mathcal{X}'$. The obtained p-values and the acceptance rate are also plotted in Figure 4 and Figure 5 as 'reciprocal'. In the experiments, we prefer to have smaller p-values when $P \neq P'$ and larger p-values when $P = P'$. The graphs show that, when $(\mu, \sigma) = (0, 1.3)$, estimating the inverted density ratio gives slightly smaller p-values and a significantly lower acceptance rate. On the other hand, when $(\mu, \sigma) = (0, 0.7)$, reciprocal estimation yields larger p-values and a significantly higher acceptance rate. When $(\mu, \sigma) = (0.3, 1)$ and $(\mu, \sigma) = (0, 1)$, the 'plain' and 'reciprocal' methods result in similar p-values and thus similar acceptance rates. These experimental results imply that, if we *adaptively* choose the plain and reciprocal approaches, the performance of homogeneity test may be improved.

Figure 4 showed that, when $P = P'$ (i.e., $(\mu, \sigma) = (0, 1)$), the p-values are large enough to reject the null hypothesis for both the plain and reciprocal approaches. Thus, the *type-I error* (the probability of rejecting correct null-hypotheses, i.e., two distributions are judged to be different when they are actually the same) would be sufficiently small for both approaches, as illustrated in Figure 5. Based on this observation, we propose to choose a smaller p-value between the plain and reciprocal approaches. This allows us to reduce the *type-II error* (the probability of accepting incorrect null-hypotheses, i.e., two distributions are judged to be the same when they are actually different), and thus the *power* of the test can be enhanced.

The experimental results of this adaptive method are also included in Figure 4 and Figure 5 as 'adaptive'. The results show that p-values obtained by the adaptive method are smaller than those obtained by the plain and reciprocal approaches, providing significant performance improvement when $P \neq P'$. On the other hand, smaller p-values can be problematic when $P = P'$ since the acceptance rate can be lowered. However, as the experimental results show, the p-values are still large enough to accept the null hypothesis and thus there is no critical performance degradation in this illustrative example.

A pseudo-code of the 'adaptive' LSTT method is summarized in Figure 6 and Figure 7. Although the permutation test process is computationally intensive, it can be easily parallelized using multi-processors/cores.

A MATLAB® implementation of LSTT is available from

http://sugiyama-www.cs.titech.ac.jp/˜sugi/software/LSTT/

```
Input: Two sets of samples $\mathcal{X} = \{\boldsymbol{x}_i\}_{i=1}^{n}$ and $\mathcal{X}' = \{\boldsymbol{x}'_j\}_{j=1}^{n'}$
Output: p-value $\widehat{p}$

$p_0 \longleftarrow \widehat{\mathrm{PE}}(\mathcal{X}, \mathcal{X}')$;
$p'_0 \longleftarrow \widehat{\mathrm{PE}}(\mathcal{X}', \mathcal{X})$;
For $t = 1, \ldots, T$
    Randomly split $\mathcal{X} \cup \mathcal{X}'$ into $\widetilde{\mathcal{X}}$ of size $|\mathcal{X}|$ and $\widetilde{\mathcal{X}}'$ of size $|\mathcal{X}'|$;
    $p_t \longleftarrow \widehat{\mathrm{PE}}(\widetilde{\mathcal{X}}, \widetilde{\mathcal{X}}')$;
    $p'_t \longleftarrow \widehat{\mathrm{PE}}(\widetilde{\mathcal{X}}', \widetilde{\mathcal{X}})$;
End
$p \longleftarrow \frac{1}{T} \sum_{t=1}^{T} I(p_t > p_0)$;
$p' \longleftarrow \frac{1}{T} \sum_{t=1}^{T} I(p'_t > p'_0)$;
$\widehat{p} \longleftarrow \min(p, p')$;
```

Figure 6: Pseudo code of LSTT. Pseudo code of $\widehat{\mathrm{PE}}(\mathcal{X}, \mathcal{X}')$ is given in Figure 7. $I(c)$ denotes the indicator function, i.e., $I(c) = 1$ if the condition c is true; otherwise $I(c) = 0$. When $|\widetilde{\mathcal{X}}| = |\widetilde{\mathcal{X}}'|$ (i.e., $n = n'$), $p'_t \longleftarrow \widehat{\mathrm{PE}}(\widetilde{\mathcal{X}}', \widetilde{\mathcal{X}})$ may be replaced by $p'_t \longleftarrow p_t$ since switching $\mathcal{X}$ and $\mathcal{X}'$ does not essentially affect the estimation of the Pearson divergence.

5 Maximum Mean Discrepancy

Maximum mean discrepancy (MMD; Borgwardt et al., 2006; Gretton et al., 2007) is a state-of-the-art method of homogeneity test. In this section, we review the definition of MMD and explain its basic properties. In the next section, the proposed LSTT is experimentally compared with MMD.

MMD is an *integral probability metric* (Müller, 1997) defined as

$$\mathrm{MMD}(\mathcal{H}, P, P') := \sup_{f \in \mathcal{H}} \left[\int f(\boldsymbol{x}) p(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} - \int f(\boldsymbol{x}) p'(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \right], \tag{11}$$

where $\mathcal{H} : \mathbb{R}^d \to \mathbb{R}$ is some function class. When $\mathcal{H}$ is a unit ball in a *universal reproducing kernel Hilbert space* (universal RKHS; Steinwart, 2001 defined on a compact metric space, then $\mathrm{MMD}(\mathcal{H}, P, P')$ vanishes if and only if $P = P'$. Gaussian RKHSs are examples of the universal RKHS.

Let $K(\boldsymbol{x}, \boldsymbol{x}')$ be a reproducing kernel function. Then the reproducing property (Aronszajn, 1950) allows one to extract the value of a function $f \in \mathcal{H}$ at a point $\boldsymbol{x}$ as

$$f(\boldsymbol{x}) = \langle f(\cdot), K(\boldsymbol{x}, \cdot) \rangle_{\mathcal{H}}, \tag{12}$$

where $\langle \cdot, \cdot \rangle_{\mathcal{H}}$ denotes the inner product in the RKHS $\mathcal{H}$. Let $\|\cdot\|_{\mathcal{H}}$ be the norm in the

```
**Input:** Two sets of samples $\mathcal{X} = \{\boldsymbol{x}_i\}_{i=1}^{n}$ and $\mathcal{X}' = \{\boldsymbol{x}'_j\}_{j=1}^{n'}$
**Output:** Pearson divergence estimate $\widehat{\mathrm{PE}}(\mathcal{X}, \mathcal{X}')$

Randomly split $\mathcal{X}$ into $\{\mathcal{X}_m\}_{m=1}^{M}$ and $\mathcal{X}'$ into $\{\mathcal{X}'_m\}_{m=1}^{M}$;
**For** each candidate of Gaussian width $\sigma$
    **For** $m = 1, \ldots, M$
        % $\boldsymbol{k}_\sigma(\boldsymbol{x}) = (1, K_\sigma(\boldsymbol{x}, \boldsymbol{x}_1), \ldots, K_\sigma(\boldsymbol{x}, \boldsymbol{x}_n))^\top$
        % $K_\sigma(\boldsymbol{x}, \boldsymbol{x}') = \exp\left(-\frac{\|\boldsymbol{x}-\boldsymbol{x}'\|^2}{2\sigma^2}\right)$
        $\widehat{\boldsymbol{G}}_m \longleftarrow \sum_{\boldsymbol{x}' \in \mathcal{X}'_m} \boldsymbol{k}_\sigma(\boldsymbol{x}')\boldsymbol{k}_\sigma(\boldsymbol{x}')^\top$;
        $\widehat{\boldsymbol{g}}_m \longleftarrow \sum_{\boldsymbol{x} \in \mathcal{X}_m} \boldsymbol{k}_\sigma(\boldsymbol{x})$;
    **End**
    **For** each candidate of regularization parameter $\lambda$
        **For** $m = 1, \ldots, M$
            $\widehat{\boldsymbol{\alpha}}_m \longleftarrow \left(\frac{1}{|\mathcal{X}' \backslash \mathcal{X}'_m|} \sum_{m' \neq m} \widehat{\boldsymbol{G}}_{m'} + \lambda \boldsymbol{I}_{n+1}\right)^{-1} \left(\frac{1}{|\mathcal{X} \backslash \mathcal{X}_m|} \sum_{m' \neq m} \widehat{\boldsymbol{g}}_{m'}\right)$;
            $\widehat{J}_m^{\mathrm{CV}}(\sigma, \lambda) \longleftarrow \frac{1}{2|\mathcal{X}'_m|} \widehat{\boldsymbol{\alpha}}_m^\top \widehat{\boldsymbol{G}}_m \widehat{\boldsymbol{\alpha}}_m - \frac{1}{|\mathcal{X}_m|} \widehat{\boldsymbol{\alpha}}_m^\top \widehat{\boldsymbol{g}}_m$;
        **End**
        $\widehat{J}^{\mathrm{CV}}(\sigma, \lambda) \longleftarrow \frac{1}{M} \sum_{m=1}^{M} \widehat{J}_m^{\mathrm{CV}}(\sigma, \lambda)$;
    **End**
**End**
$(\widehat{\sigma}, \widehat{\lambda}) \longleftarrow \operatorname*{argmin}_{(\sigma, \lambda)} \widehat{J}^{\mathrm{CV}}(\sigma, \lambda)$;
$\widehat{\boldsymbol{h}} \longleftarrow \frac{1}{|\mathcal{X}|} \sum_{\boldsymbol{x} \in \mathcal{X}} \boldsymbol{k}_{\widehat{\sigma}}(\boldsymbol{x})$;
$\widehat{\boldsymbol{\alpha}} \longleftarrow \left(\frac{1}{|\mathcal{X}'|} \sum_{\boldsymbol{x}' \in \mathcal{X}'} \boldsymbol{k}_{\widehat{\sigma}}(\boldsymbol{x}')\boldsymbol{k}_{\widehat{\sigma}}(\boldsymbol{x}')^\top + \widehat{\lambda} \boldsymbol{I}_{n+1}\right)^{-1} \widehat{\boldsymbol{h}}$;
$\widehat{\mathrm{PE}}(\mathcal{X}, \mathcal{X}') \longleftarrow \frac{1}{2} \widehat{\boldsymbol{\alpha}}^\top \widehat{\boldsymbol{h}} - \widehat{\boldsymbol{\alpha}}^\top \left(\frac{1}{|\mathcal{X}'|} \sum_{\boldsymbol{x}' \in \mathcal{X}'} \boldsymbol{k}_{\widehat{\sigma}}(\boldsymbol{x}')\right) + \frac{1}{2}$;
```

Figure 7: Pseudo code of uLSIF-based Pearson divergence estimator.

RKHS $\mathcal{H}$. Then, one can explicitly express MMD in terms of the kernel function as

$$\begin{aligned}\mathrm{MMD}(\mathcal{H}, P, P') &= \sup_{\|f\|_{\mathcal{H}}\leq 1}\left[\int \langle f(\cdot), K(\boldsymbol{x}, \cdot)\rangle_{\mathcal{H}}\, p(\boldsymbol{x})\mathrm{d}\boldsymbol{x} - \int \langle f(\cdot), K(\boldsymbol{x}, \cdot)\rangle_{\mathcal{H}}\, p'(\boldsymbol{x})\mathrm{d}\boldsymbol{x}\right]\\ &= \sup_{\|f\|_{\mathcal{H}}\leq 1}\left\langle f(\cdot), \int K(\boldsymbol{x}, \cdot)p(\boldsymbol{x})\mathrm{d}\boldsymbol{x} - \int K(\boldsymbol{x}, \cdot)p'(\boldsymbol{x})\mathrm{d}\boldsymbol{x}\right\rangle_{\mathcal{H}}\\ &= \left\|\int K(\boldsymbol{x}, \cdot)p(\boldsymbol{x})\mathrm{d}\boldsymbol{x} - \int K(\boldsymbol{x}, \cdot)p'(\boldsymbol{x})\mathrm{d}\boldsymbol{x}\right\|_{\mathcal{H}},\end{aligned}$$

where the *Cauchy-Schwarz inequality* (Bachman & Narici, 2000) was used in the last equality. Furthermore, by using

$$K(\boldsymbol{x}, \boldsymbol{x}') = \langle K(\boldsymbol{x}, \cdot), K(\boldsymbol{x}', \cdot)\rangle_{\mathcal{H}},$$

the squared MMD can be expressed as

$$\begin{aligned}\mathrm{MMD}^2(\mathcal{H}, P, P') &= \left\|\int K(\boldsymbol{x}, \cdot)p(\boldsymbol{x})\mathrm{d}\boldsymbol{x} - \int K(\boldsymbol{x}, \cdot)p'(\boldsymbol{x})\mathrm{d}\boldsymbol{x}\right\|_{\mathcal{H}}^2\\ &= \iint K(\boldsymbol{x}, \boldsymbol{x}')p(\boldsymbol{x})p(\boldsymbol{x}')\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{x}' + \iint K(\boldsymbol{x}, \boldsymbol{x}')p'(\boldsymbol{x})p'(\boldsymbol{x}')\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{x}'\\ &\quad - 2\iint K(\boldsymbol{x}, \boldsymbol{x}')p(\boldsymbol{x})p'(\boldsymbol{x})\mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{x}'.\end{aligned}$$

The above expression allows one to immediately obtain an empirical estimator—with the i.i.d. samples $\mathcal{X} = \{\boldsymbol{x}_i\}_{i=1}^{n}$ following $p(\boldsymbol{x})$ and $\mathcal{X}' = \{\boldsymbol{x}'_j\}_{j=1}^{n'}$ following $p'(\boldsymbol{x})$, a consistent estimator of $\mathrm{MMD}^2(\mathcal{H}, P, P')$ is given as

$$\begin{aligned}\widehat{\mathrm{MMD}}^2(\mathcal{H}, \mathcal{X}, \mathcal{X}') := \frac{1}{n^2}\sum_{i,i'=1}^{n} K(\boldsymbol{x}_i, \boldsymbol{x}_{i'}) + \frac{1}{n'^2}\sum_{j,j'=1}^{n'} K(\boldsymbol{x}'_j, \boldsymbol{x}'_{j'})\\ - \frac{2}{nn'}\sum_{i=1}^{n}\sum_{j=1}^{n'} K(\boldsymbol{x}_i, \boldsymbol{x}'_j).\end{aligned}$$

By the same permutation test procedure as the one described in Section 4.1, one can compute p-values for $\widehat{\mathrm{MMD}}^2(\mathcal{H}, \mathcal{X}, \mathcal{X}')$. Furthermore, an asymptotic distribution of $\widehat{\mathrm{MMD}}^2(\mathcal{H}, \mathcal{X}, \mathcal{X}')$ under $P = P'$ can be explicitly obtained (Borgwardt et al., 2006; Gretton et al., 2007). This allows one to compute the p-values without resorting to the computationally-intensive permutation procedure, which is an advantage of MMD over LSTT.

$\widehat{\mathrm{MMD}}^2(\mathcal{H}, \mathcal{X}, \mathcal{X}')$ depends on the choice of the universal RKHS $\mathcal{H}$. In the original MMD papers (Borgwardt et al., 2006; Gretton et al., 2007), the Gaussian RKHS with width set to the median distance between samples was used, which is a popular heuristic in the kernel method community (Schölkopf & Smola, 2002). Recently, an idea of using the universal RKHS yielding the maximum MMD value has been introduced (Sriperumbudur et al., 2009). In the experiments in the next section, we use this maximum-MMD technique for choosing the universal RKHS, which we confirmed to work better than the 'median' heuristic.

6 Experiments

In this section, we report experimental results comparing the performance of the proposed LSTT (Section 4) with that of the state-of-the-art MMD (Section 5).

6.1 IDA Benchmark Datasets

In the first set of experiments, we used binary classification datasets taken from the *IDA repository* (Rätsch et al., 2001). For each dataset, we randomly split all the positive training samples into two disjoint sets, $\mathcal{X}$ and $\mathcal{X}'$ with $|\mathcal{X}| = |\mathcal{X}'|$.

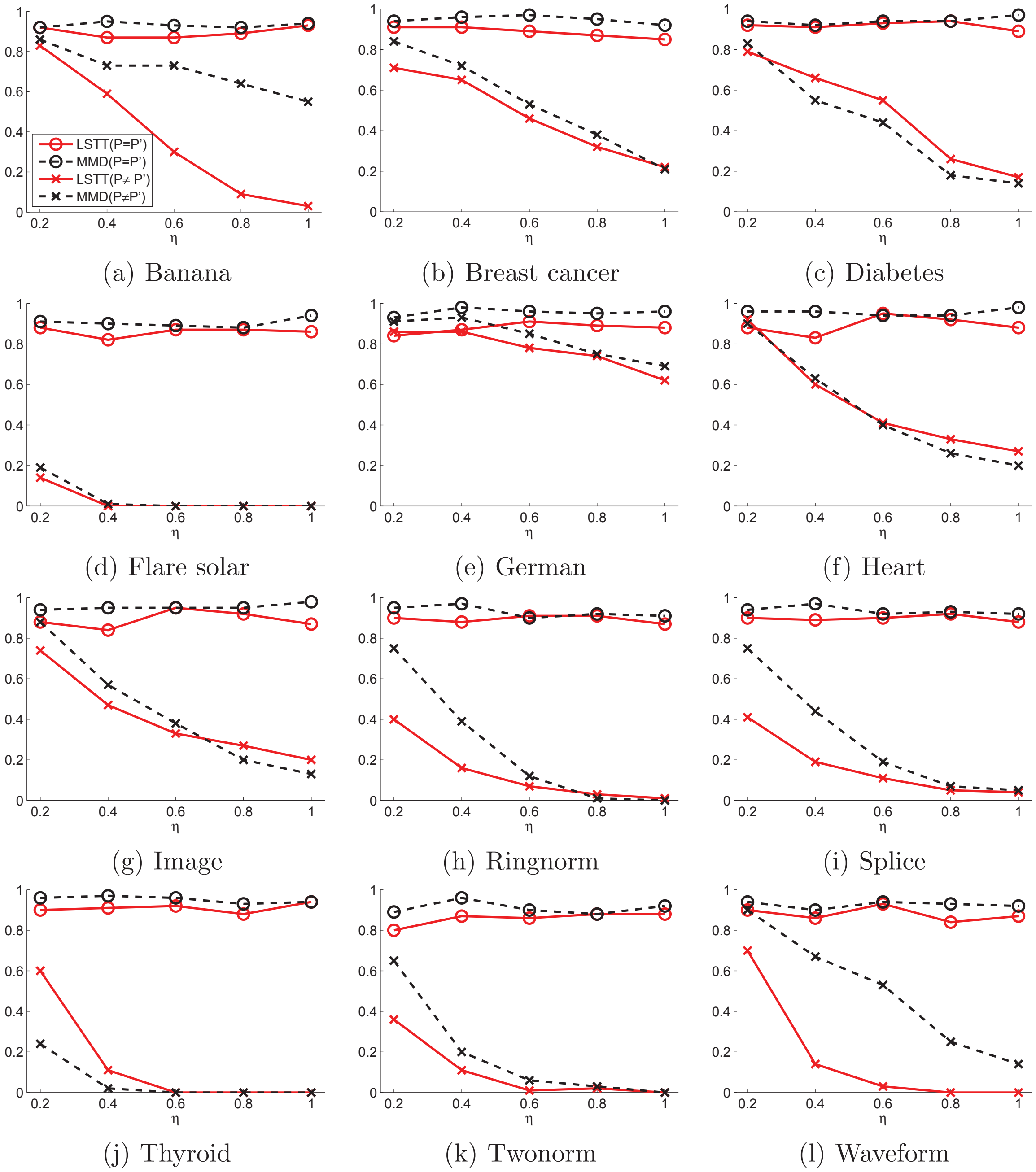
We first investigated whether the tests can correctly accept the null hypotheses (i.e., $\mathcal{X}$ and $\mathcal{X}'$ follow the same distribution). We used the Gaussian kernel both for LSTT and MMD. The Gaussian width and the regularization parameter in LSTT were determined by 5-fold cross-validation (see Section 2.4). The Gaussian width in MMD was chosen so that the MMD value is maximized (see Section 5). Since the permutation test procedures in LSTT and MMD are exactly the same, we are purely comparing the performance of the MMD and LSTT criteria in this experiment.

We investigated the rate of accepting the null hypothesis as functions of the relative sample size η for the significance level 0.05. The relative sample size η means that we used samples of size $\eta|\mathcal{X}|$ and $\eta|\mathcal{X}'|$ for homogeneity test. The experimental results are plotted in Figure 8 by lines with '$\circ$'-symbols. The results show that both methods almost always accepted the null hypothesis correctly, meaning that the type-I error is small enough for both MMD and LSTT. However, MMD seems to perform slightly better than LSTT in terms of the type-I error.

Next, we replaced a fraction of samples in the set $\mathcal{X}'$ by randomly chosen negative training samples. Thus, while $\mathcal{X}$ contains only positive training samples, $\mathcal{X}'$ includes both positive and negative training samples. The experimental results are also plotted in Figure 8 by lines with '$\times$'-symbols. The results show that LSTT tended to correctly reject the null hypothesis more frequently than MMD for the 'banana', 'ringnorm', 'splice', 'twonorm', and 'waveform' datasets. MMD worked better than LSTT for the 'thyroid' dataset, and the two methods were comparable to each other for the other datasets. Overall, LSTT compares favorably with MMD in terms of the type-II error.

6.2 USPS Hand-Written Digit Dataset

In the second sets of experiments, we used the *USPS hand-written digit* dataset provided by U.S. Postal Service (Hastie et al., 2001). Each digit image (representing an integer in



(a) Banana (b) Breast cancer (c) Diabetes

(d) Flare solar (e) German (f) Heart

(g) Image (h) Ringnorm (i) Splice

(j) Thyroid (k) Twonorm (l) Waveform

Figure 8: The rate of accepting the null hypothesis (i.e., $P = P'$) for IDA datasets under the significance level 0.05. η indicates the relative sample size we used in the experiments.

$\{0, 1, 2, \ldots, 9\}$) consists of 256 ($= 16 \times 16$) pixels, each of which takes a value between -1 to $+1$ representing its intensity level in gray-scale.

We formed two sets of samples as follows: one consists of 500 samples randomly chosen from class c ($\in \{0, 1, 2, \ldots, 9\}$), while the other consists of $500(1 - \delta)$ samples randomly chosen from class c and 500δ samples randomly chosen from another class c' ($\neq c$), where δ is the contamination rate. The goal is to test whether the two sets of samples are drawn from the same distribution or not for various contamination rates.

Table 1 shows the number of times LSTT or MMD incorrectly rejected the null hypothesis over 10 runs when the null hypothesis is correct (i.e., $\delta = 0$, meaning that the two distributions are the same). Thus, the smaller the number is, the better the performance is. The significance level was set to 0.05. The format 'l/m' in the table means that LSTT and MMD rejected the null hypothesis l and m times, respectively. The results show that both LSTT and MMD almost always accepted the correct null hypothesis successfully.

Next, we compared the performance of LSTT and MMD when the contamination rate was increased as $\delta = 0.02, 0.04, 0.06, \ldots, 0.2$. Table 2 shows the number of times LSTT or MMD rejected the null hypothesis with a lower contamination rate δ. The format '$l/t/m$' in the table means that LSTT rejected the null hypothesis with a lower contamination rate δ than MMD l times, and vice versa for m times. t denotes the number of times the smallest δ that LSTT and MMD rejected the null hypothesis was the same. The significance level was set to 0.05. The results show that LSTT tended to reject the null hypothesis with low contamination rate δ.

6.3 Brown Corpus Dataset with Tree Kernels

In the last set of experiments, we compared the performance of LSTT and MMD using natural language datasets.

We used the *Brown corpus* dataset[4], which is a carefully compiled selection of current American English. The Brown corpus consists of a million words sampled from 15 genres such as news and religion, and it is accompanied with *part-of-speech* tags, which represent relationship with adjacent and related words in a phrase, sentence, or paragraph. We converted the Brown corpus data to *dependency tree* representation by the *MaltParser*[5].

We prepared two sets of dependency trees as follows: one consists of 1000 samples taken from the 'news' category, and the other consists of $1000(1 - \delta)$ samples taken from the 'news' category and 1000δ samples taken from the 'romance' category, where δ is the contamination rate. The goal is to test whether the two sets of samples were drawn from the same distribution or not for various contamination rates.

We computed the *labeled ordered tree kernel* (Kashima & Koyanagi, 2002) between two dependency trees, which counts the number of sub-trees common to both trees. Then

[4] The Brown corpus dataset can be downloaded by using the *Natural Language Toolkit*, which contains open source Python modules, linguistic data, and documentation for research and development in natural language processing and text analysis. The Natural Language Toolkit is available from http://www.nltk.org/.

[5] The MaltParser is available from http://maltparser.org/.

Table 1: The experimental results for the USPS datasets. The number of times LSTT or MMD incorrectly rejected the null hypothesis over 10 runs when the null hypothesis was correct (i.e., the two distributions are the same). c in the table denotes the target class. The format 'l/m' means that LSTT and MMD rejected the null hypothesis l and m times, respectively. The significance level was set to 0.05.

c	0	1	2	3	4	5	6	7	8	9
	0/1	1/0	1/0	0/0	0/1	1/0	1/0	0/1	0/0	1/0

Table 2: The experimental results for the USPS datasets. The number of times LSTT or MMD rejected the null hypothesis with a smaller contamination rate. c denotes the target class and c' denotes the contamination class. The format '$l/t/m$' means that LSTT/MMD rejected the null hypothesis with a smaller contamination rate than MMD/LSTT l/m times, while the smallest contamination rate that LSTT and MMD rejected the null hypothesis was the same t times. The significance level was set to 0.05. The numbers are boldfaced if they are larger than or equal to 5.

$c\backslash c'$	0	1	2	3	4	5	6	7	8	9
0	–	**6**/2/2	**6**/3/1	**5**/1/4	**6**/2/2	**5**/2/3	**7**/2/1	**6**/1/3	**7**/0/3	**6**/3/1
1	**5**/4/1	–	4/3/3	3/2/**5**	3/1/**6**	3/2/**5**	3/4/3	3/1/**6**	2/3/**5**	3/1/**6**
2	2/**6**/2	2/2/**6**	–	2/1/**7**	3/4/3	2/1/**7**	4/0/**6**	2/0/**8**	3/1/**6**	**5**/2/3
3	**7**/3/0	4/4/2	**9**/1/0	–	**6**/4/0	1/3/**6**	**10**/0/0	1/**5**/4	**9**/1/0	4/**5**/1
4	**8**/1/1	3/2/**5**	**7**/2/1	**8**/0/2	–	**7**/1/2	**7**/2/1	2/3/**5**	**8**/1/1	4/2/4
5	**6**/3/1	**7**/2/1	**9**/1/0	4/2/4	**5**/4/1	–	**6**/3/1	4/3/3	**8**/1/1	**7**/3/0
6	**6**/2/2	**8**/2/0	**9**/1/0	**8**/1/1	**8**/1/1	**6**/2/2	–	**8**/2/0	**8**/2/0	**9**/1/0
7	**7**/2/1	**6**/2/2	**7**/1/2	**7**/1/2	**7**/0/3	**6**/2/2	**7**/1/2	–	**7**/1/2	**7**/0/3
8	**5**/3/2	3/1/**6**	**7**/1/2	**5**/2/3	3/4/3	4/3/3	**7**/1/2	3/2/**5**	–	**5**/1/4
9	**8**/1/1	**6**/3/1	**8**/0/2	**9**/0/1	4/2/4	**8**/0/2	**8**/1/1	1/1/**8**	**9**/0/1	–

the kernel values were directly fed into the LSTT and MMD algorithms. The labeled ordered tree kernel contains the *decay factor* parameter γ $(0 < \gamma \leq 1)$, which controls the weights for large sub-trees (Collins & Duffy, 2002). We computed kernel values for $\gamma = 0.1, 0.4, 0.7$, and chose the one that minimized the cross-validation score in the case of LSTT and the one that maximized the MMD value in the case of MMD.

We first investigated the number of times LSTT or MMD incorrectly rejected the null hypothesis when the null hypothesis was correct (i.e., $\delta = 0$, meaning that the two distributions are the same). Thus, the smaller the number is, the better the performance is. The significance level was set to 0.05. The results were that LSTT rejected the correct null hypothesis 30 times out of 100 runs, while MMD rejected the correct null hypothesis only 8 times. Thus MMD gave smaller type-I error.

Next, we compared the performance of LSTT and MMD when the contamination rate was increased as $\delta = 0.05, 0.1, 0.15, \ldots, 0.35$. The significance level was set to 0.05. The results were that LSTT rejected the null hypothesis with a lower contamination rate δ

than MMD 60 times out of 100 runs, while MMD rejected the null hypothesis with a lower contamination rate δ than MMD only 18 times; The smallest δ that LSTT and MMD rejected the null hypothesis was the same 22 times. This means that LSTT tended to reject the null hypothesis with low contamination rate δ.

7 Conclusions

We proposed a novel method of testing homogeneity called the *least-squares two-sample test* (LSTT). Through various experiments, we overall confirmed that LSTT tends to produce smaller type-II error than the state-of-the-art MMD method, with slightly larger type-I error.

The performance of LSTT relies on the accuracy of density ratio estimation. We adopted *unconstrained least-squares importance fitting* (uLSIF; Kanamori et al., 2009a) since it possesses the optimal non-parametric convergence rate and optimal numerical stability (Kanamori et al., 2009b). uLSIF is computationally highly efficient thanks to the analytic-form solution, which is an attractive feature in the computationally-demanding permutation test procedure. Nevertheless, the permutation test procedure is still time consuming, so speedup is an important future research topic.

We have elucidated the convergence rate of our uLSIF-based Pearson divergence estimator. We further showed that our uLSIF-based Pearson divergence estimator even achieves a faster convergence rate when the two distributions are the same. An important future study along this line of research is to elucidate the asymptotic distribution of the LSTT estimator so that homogeneity testing can be carried out analytically.

Based on the uLSIF estimator $\widehat{r}(\boldsymbol{x})$, we constructed a *consistent* Pearson divergence estimator given by

$$\widehat{\mathrm{PE}}(\mathcal{X},\mathcal{X}') := \frac{1}{2n}\sum_{i=1}^{n}\widehat{r}(\boldsymbol{x}_i) - \frac{1}{n'}\sum_{j=1}^{n'}\widehat{r}(\boldsymbol{x}'_j) + \frac{1}{2}.$$

On the other hand, it is possible to construct different consistent estimators, e.g.,

$$\widehat{\mathrm{PE}}'(\mathcal{X},\mathcal{X}') := \frac{1}{2n}\sum_{i=1}^{n}\widehat{r}(\boldsymbol{x}_i) - \frac{1}{2},$$

$$\widehat{\mathrm{PE}}''(\mathcal{X},\mathcal{X}') := -\frac{1}{2n'}\sum_{j=1}^{n'}\widehat{r}(\boldsymbol{x}'_j)^2 + \frac{1}{n}\sum_{i=1}^{n}\widehat{r}(\boldsymbol{x}_i) - \frac{1}{2}.$$

$\widehat{\mathrm{PE}}'(\mathcal{X},\mathcal{X}')$ would be the simplest estimator, while $\widehat{\mathrm{PE}}''(\mathcal{X},\mathcal{X}')$ can be obtained as the *Legendre-Fenchel dual* of the Pearson divergence (Nguyen et al., 2010). Investigating theoretical and experimental performance of these variants in terms of accuracy and computational efficiency is left open as a future work.

Recently, novel approaches to density ratio estimation for high-dimensional problems have been explored (Sugiyama et al., 2010; Yamada et al., 2010; Sugiyama et al., 2011).

In our future work, we would like to incorporate these new ideas into the framework of LSTT and see how the test performance can be improved.

Acknowledgment

MS was supported by SCAT, AOARD, and the JST PRESTO program. TS was supported by MEXT Grant-in-Aid for Young Scientists (B) 22700289. TK was supported by MEXT Grant-in-Aid for Young Scientists 20700251. MK was supported by the JST PRESTO program.

A Proof of Theorem 1

In this section, we prove Theorem 1. For simplicity we consider a situation where $n = n'$. Even if $n \neq n'$, the following proof is valid for $n := \min(n, n')$.

For arbitrary function f, let

$$P_n f := \frac{1}{n}\sum_{i=1}^{n} f(\boldsymbol{x}_i), \qquad P'_n f := \frac{1}{n}\sum_{j=1}^{n} f(\boldsymbol{x}'_j),$$

$$P f := \mathrm{E}_{\boldsymbol{x}\sim p'}[f(\boldsymbol{x})], \qquad P' f := \mathrm{E}_{\boldsymbol{x}'\sim p}[f(\boldsymbol{x}')].$$

Let $\mathcal{G}$ be a reproducing kernel Hilbert space (RKHS) corresponding to a kernel $K : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$, and the estimated density ratio $\widetilde{g}$ is defined as the minimizer of the following minimization problem:

$$\widetilde{g} := \mathop{\arg\min}_{g\in\mathcal{G}} \frac{1}{2n}\sum_{j=1}^{n} g(\boldsymbol{x}'_j)^2 - \frac{1}{n}\sum_{i=1}^{n} g(\boldsymbol{x}_i) + \frac{\lambda_n}{2}\|g\|_{\mathcal{G}}^2.$$

The estimated Pearson divergence $\widehat{\mathrm{PE}}$ is computed as

$$\widehat{\mathrm{PE}}(\mathcal{X}, \mathcal{X}') = \frac{1}{2}P_n\widetilde{g} - P'_n\widetilde{g} + \frac{1}{2}.$$

By Mercer's theorem, the kernel $K(\boldsymbol{x}, \boldsymbol{x}')$ has the following spectrum decomposition with respect to p':

$$K(\boldsymbol{x}, \boldsymbol{x}') = \sum_{k=1}^{\infty} e_k(\boldsymbol{x})\mu_k e_k(\boldsymbol{x}'),$$

where $\{e_k\}_{k=1}^{\infty}$ is an orthogonal system in $L_2(p')$, i.e., $Pe_k^2 = 1$ and $P(e_k e_{k'}) = 0$ for $k \neq k'$. Define $\mathcal{N}(\lambda)$ as

$$\mathcal{N}(\lambda) := \sum_{k=1}^{\infty} \frac{\mu_k}{\mu_k + \lambda}.$$

We assume the following conditions:

- $\sup_{\boldsymbol{x}\in\mathbb{R}^d} K(\boldsymbol{x},\boldsymbol{x}) \leq 1$,
- The constant function 1 is contained in $\mathcal{G}$: $1 \in \mathcal{G}$,
- The true density ratio p/p' is contained in $\mathcal{G}$: $p/p' = g^* \in \mathcal{G}$,
- There exists a constant $0 < \gamma < 1$ such that the spectrum μ_k of the kernel decays as $\mu_k \leq ck^{-\frac{2}{\gamma}}$, where c is a positive constant.

Then we obtain the following theorem and lemma (these are more precise versions of Theorem 1).

Theorem 1'. *Under the assumption described above, for* $\lambda_n = \left(\frac{\log n}{n}\right)^{2/(2+\gamma)}$, *we have*

$$|\widehat{\mathrm{PE}}(\mathcal{X},\mathcal{X}') - \mathrm{PE}(P,P')| = \mathcal{O}_p\left(\left(\frac{\log n}{n}\right)^{\frac{2}{2+\gamma}} + \sqrt{P'(g^*-1)^2}\left(\frac{\log n}{n}\right)^{\frac{1}{2+\gamma}}\right).$$

Lemma 1. *Suppose that the assumption described above hold and*

$$n \geq \frac{64\log^2(12/\eta)\mathcal{N}(\lambda_n)}{\lambda_n}. \tag{13}$$

Then we have

$$\begin{aligned}
&|\widehat{\mathrm{PE}}(\mathcal{X},\mathcal{X}') - \mathrm{PE}(P,P')| \\
&\leq 8\log(12/\eta)^2\left(\frac{16}{n^2\lambda_n} + \frac{(\|g^*\|_{\mathcal{G}} + \sqrt{\|g^*\|_{\mathcal{G}}} + \|g^*\|_{\mathcal{G}}^{\frac{3}{2}})\mathcal{N}(\lambda_n)}{n}\right) \\
&+ \log(12/\eta)\left(\frac{4\|g^*\|_{\mathcal{G}}}{n} + (\|g^*\|_{\mathcal{G}} + \|g^*\|_{\mathcal{G}}^{\frac{3}{2}})\sqrt{\frac{\lambda_n\mathcal{N}(\lambda_n)}{n}}\right) \\
&+ \frac{3}{2}\lambda_n C_{n,\eta} \\
&+ \log(12/\eta)\left(\frac{4\|g^*-1\|_\infty}{n} + \sqrt{\frac{P'(g^*-1)^2}{n}} + \sqrt{\frac{P(g^*-1)^2}{n}}\right) \\
&+ \frac{1}{2}\sqrt{P'(g^*-1)^2}\sqrt{128\log^2(12/\eta)\left(\frac{8}{n^2\lambda_n} + \frac{(\|g^*\|_{\mathcal{G}} + \|g^*\|_{\mathcal{G}}^2)\mathcal{N}(\lambda_n)}{n}\right) + 2\lambda_n\|g^*\|_{\mathcal{G}}^2},
\end{aligned} \tag{14}$$

with probability at least $1-\eta$, *where*

$$C_{n,\eta} = \frac{\|1\|_{\mathcal{G}}^2}{1+\lambda_n\|1\|_{\mathcal{G}}^2}\left\{8\left(\|g^*\|_{\mathcal{G}}^2 + 8\log^2(12/\eta)\left(\frac{4}{\lambda_n^2 n^2} + \frac{\mathcal{N}(\lambda_n)\|g^*\|_{\mathcal{G}}}{\lambda_n n}\right)\right) + 1\right\}.$$

Before proving the lemma, we introduce the following proposition that is a part of Proposition 2 in Caponnetto and de Vito (2007).

Proposition 1. *Let ξ be a random variable taking values in a real separable Hilbert space $\mathcal{K}$ on a probability space $(\Omega, \mathcal{F}, P)$. Assume that there are two positive constants L and σ such that*

$$\|\xi\|_{\mathcal{K}} \leq \frac{L}{2} \quad a.s., \tag{15}$$

$$\mathrm{E}[\|\xi\|_{\mathcal{K}}^2] \leq \sigma^2. \tag{16}$$

Then, for all $n \geq 1$ and $0 < \eta < 1$, it holds that

$$\mathrm{Prob}_{(\omega_1,\dots,\omega_n)\sim P^n}\left[\left\|\frac{1}{n}\sum_{i=1}^{n}\xi(\omega_i) - \mathrm{E}[\xi]\right\|_{\mathcal{K}} \leq 2\left(\frac{L}{n} + \frac{\sigma}{\sqrt{n}}\right)\log\frac{2}{\eta}\right] \geq 1-\eta. \tag{17}$$

Proof of Lemma 1. First we define some notation. Let $K_{\boldsymbol{x}}$ be an element of $\mathcal{G}$ such that

$$\langle K_{\boldsymbol{x}}, f\rangle = f(\boldsymbol{x})$$

for $f \in \mathcal{G}$ and $\boldsymbol{x} \in \mathbb{R}^d$, i.e., $K_{\boldsymbol{x}}(\cdot) = K(\boldsymbol{x}, \cdot)$ as an element of $\mathcal{G}$. We define $T_{p'} : \mathcal{G} \to \mathcal{G}$ as

$$\langle g, T_{p'}f\rangle = \mathrm{E}_{\boldsymbol{x}'\sim p'}[g(\boldsymbol{x}')f(\boldsymbol{x}')],$$

for $f, g \in \mathcal{G}$. Similarly we define $\widehat{T}_{p'} : \mathcal{G} \to \mathcal{G}$ as

$$\langle g, \widehat{T}_{p'}f\rangle = \frac{1}{n}\sum_{j=1}^{n} g(\boldsymbol{x}_j')f(\boldsymbol{x}_j').$$

Note that $T_{p'} = \mathrm{E}_{\boldsymbol{x}'\sim p'}[K_{\boldsymbol{x}'}K_{\boldsymbol{x}'}^{\circ}]$ where $K_{\boldsymbol{x}}^{\circ}$ is the adjoint of $K_{\boldsymbol{x}}$. Let $\phi_k := \sqrt{\mu_k}e_k$. Then $\{\phi_k\}_{k=1}^{\infty}$ is a complete orthonormal system in the RKHS $\mathcal{G}$, and $T_{p'}$ can be represented as

$$T_{p'} = \sum_{k=1}^{\infty}\phi_k\mu_k\phi_k^{\circ}.$$

Let $h_1, \widehat{h}_1, h_2, \widehat{h}_2 \in \mathcal{G}$ be

$$h_1 := \mathrm{E}_{\boldsymbol{x}'\sim p'}[K_{\boldsymbol{x}'}], \quad \widehat{h}_1 = \frac{1}{n}\sum_{j=1}^{n} K_{\boldsymbol{x}_j'},$$

$$h_2 := \mathrm{E}_{\boldsymbol{x}\sim p}[K_{\boldsymbol{x}}] = \mathrm{E}_{\boldsymbol{x}'\sim p'}[K_{\boldsymbol{x}'}g^*(\boldsymbol{x}')] = \mathrm{E}_{\boldsymbol{x}'\sim p'}[K_{\boldsymbol{x}'}\langle K_{\boldsymbol{x}'}, g^*\rangle_{\mathcal{G}}] = T_{p'}g^*, \quad \widehat{h}_2 = \frac{1}{n}\sum_{i=1}^{n} K_{\boldsymbol{x}_i}.$$

Note that $\mathrm{E}[\widehat{h}_1] = h_1$ and $\mathrm{E}[\widehat{h}_2] = h_2$, and

$$\langle h_1, f\rangle = P'f, \quad \langle \widehat{h}_1, f\rangle = P_n'f, \quad \langle h_2, f\rangle = Pf, \quad \langle \widehat{h}_2, f\rangle = P_nf. \tag{18}$$

It can be easily checked that

$$\widetilde{g} = (\widehat{T}_{p'} + \lambda_n)^{-1}\widehat{h}_2.$$

Here we define

$$g_{\lambda_n} = (T_{p'} + \lambda_n)^{-1} h_2.$$

The difference between $\widehat{\mathrm{PE}}(\mathcal{X}, \mathcal{X}')$ and $\mathrm{PE}(P, P')$ is expanded as

$$\begin{aligned}
&\widehat{\mathrm{PE}}(\mathcal{X}, \mathcal{X}') - \mathrm{PE}(P, P') \\
&= \frac{1}{2}(P_n\widetilde{g} - Pg^*) - (P'_n\widetilde{g} - P'g^*) \\
&= \frac{1}{2}[(P_n - P)(\widetilde{g} - g^*) + P(\widetilde{g} - g^*) + (P_n - P)g^*] - (P'_n\widetilde{g} - 1). \qquad (19)
\end{aligned}$$

Since $P(\widetilde{g} - g^*)$ is bounded as

$$\begin{aligned}
|P(\widetilde{g} - g^*)| &= |P'(\widetilde{g} - g^*)| + |P'((g^* - 1)(\widetilde{g} - g^*))| \\
&\leq |P'(\widetilde{g} - g^*)| + \sqrt{P'(g^* - 1)^2}\sqrt{P'(\widetilde{g} - g^*)^2} \\
&= |(P' - P'_n)(\widetilde{g} - g^*) + P'_n\widetilde{g} - P'g^* + (P' - P'_n)g^*| \\
&\quad + \sqrt{P'(g^* - 1)^2}\sqrt{P'(\widetilde{g} - g^*)^2} \\
&\leq |(P' - P'_n)(\widetilde{g} - g^*)| + |P'_n\widetilde{g} - 1| + |(P' - P'_n)g^*| \\
&\quad + \sqrt{P'(g^* - 1)^2}\sqrt{P'(\widetilde{g} - g^*)^2}, \qquad (20)
\end{aligned}$$

Eq.(19) indicates

$$\begin{aligned}
|\widehat{\mathrm{PE}}(\mathcal{X}, \mathcal{X}') - \mathrm{PE}(P, P')| \leq &\frac{1}{2}|(P'_n - P')(\widetilde{g} - g^*)| + \frac{1}{2}|(P_n - P)(\widetilde{g} - g^*)| \\
&+ \frac{3}{2}|P'_n\widetilde{g} - 1| \\
&+ \frac{1}{2}|(P'_n - P')g^*| + \frac{1}{2}|(P_n - P)g^*| \\
&+ \frac{1}{2}\sqrt{P'(g^* - 1)^2}\sqrt{P'(\widetilde{g} - g^*)^2}. \qquad (21)
\end{aligned}$$

Step 1. Bounding $(P'_n - P')(\widetilde{g} - g^*)$

$$\begin{aligned}
&(P'_n - P')(\widetilde{g} - g^*) \\
&= \langle \widehat{h}_1 - h_1, (\widehat{T}_{p'} + \lambda_n)^{-1}\widehat{h}_2 - g^* \rangle \\
&= \langle \widehat{h}_1 - h_1, (\widehat{T}_{p'} + \lambda_n)^{-1}(\widehat{h}_2 - h_2) + (\widehat{T}_{p'} + \lambda_n)^{-1}h_2 - (T_{p'} + \lambda_n)^{-1}h_2 + (T_{p'} + \lambda_n)^{-1}h_2 - g^* \rangle \\
&= \langle \widehat{h}_1 - h_1, (\widehat{T}_{p'} + \lambda_n)^{-1}(\widehat{h}_2 - h_2) \rangle + \langle \widehat{h}_1 - h_1, (\widehat{T}_{p'} + \lambda)^{-1}(T_{p'} - \widehat{T}_{p'})(T_{p'} + \lambda_n)^{-1}h_2 \rangle \\
&\quad + \langle \widehat{h}_1 - h_1, (T_{p'} + \lambda_n)^{-1}h_2 - g^* \rangle \\
&= \underbrace{\langle \widehat{h}_1 - h_1, (\widehat{T}_{p'} + \lambda_n)^{-1}(\widehat{h}_2 - h_2) \rangle}_{\text{(1-a)}} + \underbrace{\langle \widehat{h}_1 - h_1, (\widehat{T}_{p'} + \lambda_n)^{-1}(T_{p'} - \widehat{T}_{p'})(T_{p'} + \lambda_n)^{-1}h_2 \rangle}_{\text{(1-b)}} \\
&\quad - \underbrace{\langle \widehat{h}_1 - h_1, (T_{p'} + \lambda)^{-1}\lambda_n g^* \rangle}_{\text{(1-c)}}, \qquad (22)
\end{aligned}$$

where in the last inequality we used the relation $T_{p'}g^* = h_2$.

Let $\|\cdot\|_{\mathcal{L}(\mathcal{G})}$ be the operator norm of the bounded linear operator from $\mathcal{G}$ to $\mathcal{G}$. Then

$$\begin{aligned}
&\|(T_{p'}+\lambda)^{\frac{1}{2}}(\widehat{T}_{p'}+\lambda_n)^{-1}(T_{p'}+\lambda_n)^{\frac{1}{2}}\|_{\mathcal{L}(\mathcal{G})}\\
&=\left\|[(T_{p'}+\lambda_n)^{-\frac{1}{2}}(\widehat{T}_{p'}+\lambda_n-T_{p'}-\lambda_n+T_{p'}+\lambda_n)(T_{p'}+\lambda_n)^{-\frac{1}{2}}]^{-1}\right\|_{\mathcal{L}(\mathcal{G})}\\
&=\left\|[(T_{p'}+\lambda_n)^{-\frac{1}{2}}(\widehat{T}_{p'}-T_{p'})(T_{p'}+\lambda_n)^{-\frac{1}{2}}+I]^{-1}\right\|_{\mathcal{L}(\mathcal{G})}. \qquad (23)
\end{aligned}$$

We define $\mathcal{A}_1$ as follows:

$$\mathcal{A}_1=\left\{\left\|(T_{p'}-\widehat{T}_{p'})(T_{p'}+\lambda_n)^{-1}\right\|_{\mathcal{L}(\mathcal{G})}\leq\frac{1}{2}\right\}.$$

Caponnetto and de Vito (2007) showed that under the event $\mathcal{A}_1$,

$$\left\|(T_{p'}+\lambda_n)^{-\frac{1}{2}}(\widehat{T}_{p'}-T_{p'})(T_{p'}+\lambda_n)^{-\frac{1}{2}}\right\|_{\mathcal{L}(\mathcal{G})}\leq\frac{1}{2},$$

and the probability of $\mathcal{A}_1$ is at least $1-\eta/6$ under the condition Eq.(13). Therefore we obtain

$$\begin{aligned}
&\|(T_{p'}+\lambda_n)^{\frac{1}{2}}(\widehat{T}_{p'}+\lambda_n)^{-1}(T_{p'}+\lambda_n)^{\frac{1}{2}}\|_{\mathcal{L}(\mathcal{G})}\\
&=\left\|[(T_{p'}+\lambda_n)^{-\frac{1}{2}}(\widehat{T}_{p'}-T_{p'})(T_{p'}+\lambda_n)^{-\frac{1}{2}}+I]^{-1}\right\|_{\mathcal{L}(\mathcal{G})}\\
&\leq 2 \qquad (24)
\end{aligned}$$

on the event $\mathcal{A}_1$.

Bounding (1-a):

$$\begin{aligned}
&\langle\widehat{h}_1-h_1,(\widehat{T}_{p'}+\lambda_n)^{-1}(\widehat{h}_2-h_2)\rangle\\
&\leq\langle\widehat{h}_1-h_1,(T_{p'}+\lambda_n)^{-\frac{1}{2}}[(T_{p'}+\lambda_n)^{\frac{1}{2}}(\widehat{T}_{p'}+\lambda_n)^{-1}(T_{p'}+\lambda_n)^{\frac{1}{2}}](T_{p'}+\lambda_n)^{-\frac{1}{2}}(\widehat{h}_2-h_2)\rangle\\
&\leq\left\|(T_{p'}+\lambda_n)^{-\frac{1}{2}}(\widehat{h}_1-h_1)\right\|_{\mathcal{G}}\left\|(T_{p'}+\lambda_n)^{\frac{1}{2}}(\widehat{T}_{p'}+\lambda_n)^{-1}(T_{p'}+\lambda_n)^{\frac{1}{2}}\right\|_{\mathcal{L}(\mathcal{G})}\\
&\quad\times\left\|(T_{p'}+\lambda_n)^{-\frac{1}{2}}(\widehat{h}_2-h_2)\right\|_{\mathcal{G}}.
\end{aligned}$$

According to Eq.(24), we have

$$\left\|(T_{p'}+\lambda_n)^{\frac{1}{2}}(\widehat{T}_{p'}+\lambda_n)^{-1}(T_{p'}+\lambda_n)^{\frac{1}{2}}\right\|_{\mathcal{L}(\mathcal{G})}\leq 2$$

on the event $\mathcal{A}_1$.

Let $\xi:\mathbb{R}^d\to\mathcal{G}$ be the random variable

$$\xi(\boldsymbol{x}')=(T_{p'}+\lambda_n)^{-\frac{1}{2}}K_{\boldsymbol{x}'}.$$

Then

$$\begin{aligned}(T_{p'}+\lambda_n)^{-\frac{1}{2}}(\widehat{h}_1-h_1) &= (P'_n-P')\xi\\ \|\xi\|_{\mathcal{G}} &= \sqrt{K^{\circ}_{\boldsymbol{x}'}(T_{p'}+\lambda_n)^{-1}K_{\boldsymbol{x}'}} \le \sqrt{\lambda_n^{-1}},\\ \mathrm{E}_{\boldsymbol{x}'\sim p'}[\|\xi\|^2_{\mathcal{G}}] &= \mathrm{E}_{\boldsymbol{x}'\sim p'}[K^{\circ}_{\boldsymbol{x}}(T_{p'}+\lambda_n)^{-1}K_{\boldsymbol{x}'}]\\ &= \mathrm{E}_{\boldsymbol{x}'\sim p'}\left[\sum_{k=1}^{\infty}\frac{\mu_k}{\mu_k+\lambda_n}e_k(\boldsymbol{x}')^2\right]\\ &= \sum_{k=1}^{\infty}\frac{\mu_k}{\mu_k+\lambda_n}\\ &= \mathcal{N}(\lambda_n).\end{aligned}$$

Therefore, by Proposition 1, we have

$$\|(T_{p'}+\lambda_n)^{-\frac{1}{2}}(\widehat{h}_1-h_1)\|_{\mathcal{G}} = \|(P'_n-P')\xi\|_{\mathcal{G}} \le 2\log(12/\eta)\left(\frac{2}{n\sqrt{\lambda_n}}+\sqrt{\frac{\mathcal{N}(\lambda_n)}{n}}\right), \quad (25)$$

with probability $1-\eta/6$. We define $\mathcal{A}_2$ as the event where the above inequality holds:

$$\mathcal{A}_2 := \left\{\|(T_{p'}+\lambda_n)^{-\frac{1}{2}}(\widehat{h}_1-h_1)\|_{\mathcal{G}} \le 2\log(12/\eta)\left(\frac{2}{n\sqrt{\lambda_n}}+\sqrt{\frac{\mathcal{N}(\lambda_n)}{n}}\right)\right\}. \quad (26)$$

One can obtain a similar bound for $\|(T_{p'}+\lambda_n)^{-\frac{1}{2}}(\widehat{h}_2-h_2)\|_{\mathcal{G}}$. In fact, using

$$\begin{aligned}\mathrm{E}_{\boldsymbol{x}'\sim p}\left[\sum_{k=1}^{\infty}\frac{\mu_k}{\mu_k+\lambda_n}e_k(\boldsymbol{x}')^2\right] &\le \mathrm{E}_{\boldsymbol{x}'\sim p'}\left[g^*(\boldsymbol{x}')\sum_{k=1}^{\infty}\frac{\mu_k}{\mu_k+\lambda_n}e_k(\boldsymbol{x}')^2\right]\\ &\le \|g^*\|_{\mathcal{G}}\mathrm{E}_{\boldsymbol{x}'\sim p'}\left[\sum_{k=1}^{\infty}\frac{\mu_k}{\mu_k+\lambda_n}e_k(\boldsymbol{x}')^2\right] = \|g^*\|_{\mathcal{G}}\mathcal{N}(\lambda_n), \quad (27)\end{aligned}$$

instead of Eq.(25), one can show that, by Proposition 1,

$$\|(T_{p'}+\lambda_n)^{-\frac{1}{2}}(\widehat{h}_2-h_2)\|_{\mathcal{G}} \le 2\log(12/\eta)\left(\frac{2}{n\sqrt{\lambda_n}}+\sqrt{\frac{\|g^*\|\mathcal{N}(\lambda_n)}{n}}\right), \quad (28)$$

with probability $1-\eta/6$. We define $\mathcal{A}_3$ as the event where the above inequality holds:

$$\mathcal{A}_3 := \left\{\|(T_{p'}+\lambda_n)^{-\frac{1}{2}}(\widehat{h}_2-h_2)\|_{\mathcal{G}} \le 2\log(12/\eta)\left(\frac{2}{n\sqrt{\lambda_n}}+\sqrt{\frac{\|g^*\|\mathcal{N}(\lambda_n)}{n}}\right)\right\}. \quad (29)$$

Combining Eqs.(24), (25), and (28), we can show that the term (a) is bounded as

$$|\langle\widehat{h}_1-h_1,(\widehat{T}_{p'}+\lambda_n)^{-1}(\widehat{h}_2-h_2)\rangle| \le 16\log(12/\eta)^2\left(\frac{4}{n^2\lambda_n}+\frac{\sqrt{\|g^*\|}\mathcal{N}(\lambda_n)}{n}\right), \quad (30)$$

on the event $\mathcal{A}_1\cap\mathcal{A}_2\cap\mathcal{A}_3$.

Bounding (1-b):

$$\begin{aligned}&\langle \widehat{h}_1 - h_1, (\widehat{T}_{p'} + \lambda_n)^{-1}(T_{p'} - \widehat{T}_{p'})(T_{p'} + \lambda_n)^{-1}h_2\rangle \\ &\le \|(T_{p'} + \lambda_n)^{-\frac{1}{2}}(\widehat{h}_1 - h_1)\|_{\mathcal{G}}\|(T_{p'} + \lambda_n)^{\frac{1}{2}}(\widehat{T}_{p'} + \lambda_n)^{-1}(T_{p'} + \lambda_n)^{\frac{1}{2}}\|_{\mathcal{L}(\mathcal{G})} \\ &\quad \times \|(T_{p'} + \lambda_n)^{-\frac{1}{2}}(T_{p'} - \widehat{T}_{p'})(T_{p'} + \lambda_n)^{-1}h_2\|_{\mathcal{G}}.\end{aligned}$$

We have already obtained bounds for $\|(T_{p'}+\lambda)^{-\frac{1}{2}}(\widehat{h}_1-h_1)\|$ and $\|(T_{p'}+\lambda)^{\frac{1}{2}}(\widehat{T}_{p'}+\lambda)^{-1}(T_{p'}+\lambda)^{\frac{1}{2}}\|$ in Eq.(25) and Eq.(24):

$$\|(T_{p'} + \lambda_n)^{-\frac{1}{2}}(\widehat{h}_1 - h_1)\|_{\mathcal{G}} \le 2\log(12/\eta)\left(\frac{2}{n\sqrt{\lambda_n}} + \sqrt{\frac{\mathcal{N}(\lambda_n)}{n}}\right), \quad (31)$$

$$\|(T_{p'} + \lambda_n)^{\frac{1}{2}}(\widehat{T}_{p'} + \lambda_n)^{-1}(T_{p'} + \lambda_n)^{\frac{1}{2}}\|_{\mathcal{L}(\mathcal{G})} \le 2, \quad (32)$$

on the event $\mathcal{A}_1 \cap \mathcal{A}_2$.

Let $\xi : \mathbb{R}^d \to \mathcal{G}$ be the random variable such as

$$\xi(\boldsymbol{x}) = (T_{p'} + \lambda_n)^{-\frac{1}{2}}K_{\boldsymbol{x}}K_{\boldsymbol{x}}^{\circ}(T_{p'} + \lambda_n)^{-1}h_2.$$

Then we have

$$\begin{aligned}\|\xi(\boldsymbol{x})\|_{\mathcal{G}} &= \|(T_{p'} + \lambda_n)^{-\frac{1}{2}}K_{\boldsymbol{x}}K_{\boldsymbol{x}}^{\circ}(T_{p'} + \lambda_n)^{-1}T_{p'}g^*\|_{\mathcal{G}} \\ &\le \|(T_{p'} + \lambda_n)^{-\frac{1}{2}}\|_{\mathcal{L}(\mathcal{G})}\|K_{\boldsymbol{x}}K_{\boldsymbol{x}}^{\circ}\|_{\mathcal{L}(\mathcal{G})}\|(T_{p'} + \lambda_n)^{-1}T_{p'}\|_{\mathcal{L}(\mathcal{G})}\|g^*\|_{\mathcal{G}} \\ &\le \lambda_n^{-\frac{1}{2}}\|g^*\|_{\mathcal{G}},\end{aligned}$$

where we used the relation

$$\begin{aligned}\|(K_{\boldsymbol{x}}K_{\boldsymbol{x}}^{\circ})h\|_{\mathcal{G}} &= \|K_{\boldsymbol{x}}\langle K_{\boldsymbol{x}}, h\rangle_{\mathcal{G}}\|_{\mathcal{G}} = \langle K_{\boldsymbol{x}}, h\rangle_{\mathcal{G}}\|K_{\boldsymbol{x}}\|_{\mathcal{G}} \\ &\le \|h\|_{\mathcal{G}}\|K_{\boldsymbol{x}}\|_{\mathcal{G}}^2 = \|h\|_{\mathcal{G}}K(\boldsymbol{x},\boldsymbol{x}) \le \|h\|_{\mathcal{G}}\end{aligned}$$

for all $h \in \mathcal{G}$. Then,

$$\begin{aligned}\mathrm{E}_{\boldsymbol{x}'\sim p'}[\|\xi(\boldsymbol{x}')\|_{\mathcal{G}}^2] &= \mathrm{E}_{\boldsymbol{x}'\sim p'}[\|(T_{p'} + \lambda_n)^{-\frac{1}{2}}K_{\boldsymbol{x}'}K_{\boldsymbol{x}'}^{\circ}(T_{p'} + \lambda_n)^{-1}T_{p'}g^*\|_{\mathcal{G}}^2] \\ &\le \mathrm{E}_{\boldsymbol{x}'\sim p'}\left[\|(T_{p'} + \lambda_n)^{-1}K_{\boldsymbol{x}'}K_{\boldsymbol{x}'}^{\circ}\|_{\mathcal{L}(\mathcal{G})}\right]\|K_{\boldsymbol{x}'}^{\circ}K_{\boldsymbol{x}'}\|_{\mathcal{L}(\mathcal{G})}\|(T_{p'} + \lambda_n)^{-1}T_{p'}g^*\|_{\mathcal{G}}^2 \\ &\le \mathrm{E}_{\boldsymbol{x}'\sim p'}\left[\mathrm{tr}\left((T_{p'} + \lambda_n)^{-1}K_{\boldsymbol{x}'}K_{\boldsymbol{x}'}^{\circ}\right)\right]\|g^*\|_{\mathcal{G}}^2 \\ &= \mathrm{tr}\left((T_{p'} + \lambda_n)^{-1}T_{p'}\right)\|g^*\|_{\mathcal{G}}^2 \\ &= \mathcal{N}(\lambda_n)\|g^*\|_{\mathcal{G}}^2.\end{aligned}$$

Therefore, by Proposition 1, we obtain

$$\begin{aligned}&\|(T_{p'} + \lambda_n)^{-\frac{1}{2}}(T_{p'} - \widehat{T}_{p'})(T_{p'} + \lambda_n)^{-1}h_2\|_{\mathcal{G}} \\ &\le 2\log(12/\eta)\left(\frac{2}{n\sqrt{\lambda_n}} + \sqrt{\frac{\|g^*\|^2\mathcal{N}(\lambda_n)}{n}}\right),\end{aligned} \quad (33)$$

with probability $1-\eta/6$. We define $\mathcal{A}_4$ as the event where the above inequality holds

$$\mathcal{A}_4 := \left\{ \|(T_{p'}+\lambda_n)^{-\frac{1}{2}}(T_{p'}-\widehat{T}_{p'})(T_{p'}+\lambda_n)^{-1}h_2\|_{\mathcal{G}} \right. \\ \left. \leq 2\log(12/\eta)\left(\frac{2}{n\sqrt{\lambda_n}}+\sqrt{\frac{\|g^*\|^2\mathcal{N}(\lambda_n)}{n}}\right)\right\}.$$

Combining Eqs.(25), (24), and (33), the term (1-b) is bounded as

$$|\langle \widehat{h}_1-h_1,(\widehat{T}_{p'}+\lambda_n)^{-1}(T_{p'}-\widehat{T}_{p'})(T_{p'}+\lambda_n)^{-1}h_2\rangle| \\ \leq 16\log(12/\eta)^2\left(\frac{4}{n^2\lambda_n}+\frac{\|g^*\|\mathcal{N}(\lambda_n)}{n}\right),$$

on the event $\mathcal{A}_1\cap\mathcal{A}_2\cap\mathcal{A}_4$.

Bounding (1-c): We have

$$\langle \widehat{h}_1-h_1,(T_{p'}+\lambda_n)^{-1}\lambda_n g^*\rangle = \langle (T_{p'}+\lambda_n)^{-\frac{1}{2}}(\widehat{h}_1-h_1),(T_{p'}+\lambda_n)^{-\frac{1}{2}}\lambda_n g^*\rangle \\ \leq \|(T_{p'}+\lambda_n)^{-\frac{1}{2}}(\widehat{h}_1-h_1)\|_{\mathcal{G}}\|(T_{p'}+\lambda_n)^{-\frac{1}{2}}\sqrt{\lambda_n}g^*\|_{\mathcal{G}}\sqrt{\lambda_n}. \tag{34}$$

Notice that Eq.(25) gives

$$\|(T_{p'}+\lambda_n)^{-\frac{1}{2}}(\widehat{h}_1-h_1)\|_{\mathcal{G}} \leq 2\log(12/\eta)\left(\frac{2}{n\sqrt{\lambda_n}}+\sqrt{\frac{\mathcal{N}(\lambda_n)}{n}}\right), \tag{35}$$

on the event $\mathcal{A}_2$. This and $\|(T_{p'}+\lambda_n)^{-\frac{1}{2}}\sqrt{\lambda_n}g^*\|_{\mathcal{G}} \leq \|g^*\|_{\mathcal{G}}$ give

$$|\langle \widehat{h}_1-h_1,(T_{p'}+\lambda_n)^{-1}\lambda_n g^*\rangle| \leq 2\log(12/\eta)\left(\frac{2\|g^*\|_{\mathcal{G}}}{n}+\|g^*\|_{\mathcal{G}}\sqrt{\frac{\mathcal{N}(\lambda_n)\lambda_n}{n}}\right), \tag{36}$$

on the event $\mathcal{A}_2$.

Combining the bounds of (1-a), (1-b), and (1-c):

$$\begin{aligned}
&|(P'_n - P')(\widetilde{g} - g^*)| \\
&\leq 16\log(12/\eta)^2 \left(\frac{4}{n^2\lambda_n} + \frac{\sqrt{\|g^*\|_{\mathcal{G}}}\mathcal{N}(\lambda_n)}{n} \right) + 16\log(12/\eta)^2 \left(\frac{4}{n^2\lambda_n} + \frac{\|g^*\|_{\mathcal{G}}\mathcal{N}(\lambda_n)}{n} \right) \\
&\quad + 2\log(12/\eta) \left(\frac{2\|g^*\|_{\mathcal{G}}}{n} + \|g^*\|_{\mathcal{G}}\sqrt{\frac{\mathcal{N}(\lambda_n)\lambda_n}{n}} \right) \\
&= 16\log(12/\eta)^2 \left(\frac{8}{n^2\lambda_n} + \frac{(\|g^*\|_{\mathcal{G}} + \sqrt{\|g^*\|_{\mathcal{G}}})\mathcal{N}(\lambda_n)}{n} \right) \\
&\quad + 2\log(12/\eta) \left(\frac{2\|g^*\|_{\mathcal{G}}}{n} + \|g^*\|_{\mathcal{G}}\sqrt{\frac{\mathcal{N}(\lambda_n)\lambda_n}{n}} \right),
\end{aligned}$$

on the event $\mathcal{A}_1 \cap \mathcal{A}_2 \cap \mathcal{A}_3 \cap \mathcal{A}_4$.

Step 2. Bounding $|(P_n - P)(\widetilde{g} - g^*)|$

As in Eq.(22), we have

$$\begin{aligned}
&(P_n - P)(\widetilde{g} - g^*) \\
&= \langle \widehat{h}_2 - h_2, (\widehat{T}_{p'} + \lambda_n)^{-1}(\widehat{h}_2 - h_2)\rangle + \langle \widehat{h}_2 - h_2, (\widehat{T}_{p'} + \lambda_n)^{-1}(T_{p'} - \widehat{T}_{p'})(T_{p'} + \lambda_n)^{-1}h_2\rangle \\
&\quad + \langle \widehat{h}_2 - h_2, (T_{p'} + \lambda_n)^{-1}\lambda_n g^*\rangle.
\end{aligned}$$

Using Eq.(28) instead of Eq.(25), on the event $\mathcal{A}_1 \cap \mathcal{A}_3 \cap \mathcal{A}_4$, each term is bounded as

$$\begin{aligned}
&|\langle \widehat{h}_2 - h_2, (\widehat{T}_{p'} + \lambda_n)^{-1}(\widehat{h}_2 - h_2)\rangle| \\
&\qquad \leq 16\log(12/\eta)^2 \left(\frac{4}{n^2\lambda_n} + \frac{\|g^*\|_{\mathcal{G}}\mathcal{N}(\lambda_n)}{n} \right), \\
&|\langle \widehat{h}_2 - h_2, (\widehat{T}_{p'} + \lambda_n)^{-1}(T_{p'} - \widehat{T}_{p'})(T_{p'} + \lambda_n)^{-1}h_2\rangle| \\
&\qquad \leq 16\log(12/\eta)^2 \left(\frac{4}{n^2\lambda_n} + \frac{\|g^*\|_{\mathcal{G}}^{\frac{3}{2}}\mathcal{N}(\lambda_n)}{n} \right), \\
&|\langle \widehat{h}_2 - h_2, (T_{p'} + \lambda_n)^{-1}\lambda g^*\rangle| \\
&\qquad \leq 2\log(12/\eta) \left(\frac{2\|g^*\|_{\mathcal{G}}}{n} + \|g^*\|_{\mathcal{G}}^{\frac{3}{2}}\sqrt{\frac{\mathcal{N}(\lambda_n)\lambda_n}{n}} \right).
\end{aligned}$$

Then we obtain the following bound:

$$\begin{aligned}
&|(P_n - P)(\widetilde{g} - g^*)| \\
&\leq 16 \log(12/\eta)^2 \left(\frac{8}{n^2 \lambda_n} + \frac{(\|g^*\|_{\mathcal{G}}^{\frac{3}{2}} + \|g^*\|_{\mathcal{G}}) \mathcal{N}(\lambda_n)}{n} \right) \\
&\quad + 2 \log(12/\eta) \left(\frac{2\|g^*\|_{\mathcal{G}}}{n} + \|g^*\|_{\mathcal{G}}^{\frac{3}{2}} \sqrt{\frac{\mathcal{N}(\lambda_n)}{n}} \right),
\end{aligned}$$

on the event $\mathcal{A}_1 \cap \mathcal{A}_3 \cap \mathcal{A}_4$.

Step 3. Bounding $|P'_n \widetilde{g} - 1|$

We decompose $\widetilde{g}$ as

$$\widetilde{g} = \widehat{u} + \widehat{\beta}, \tag{37}$$

where $\widehat{u} \perp 1$ in $\mathcal{G}$ and $\widehat{\beta}$ is a constant function. Then one can easily show that

$$\widehat{\beta} = \frac{1 - P'_n \widehat{u}}{1 + \lambda_n \|1\|_{\mathcal{G}}}.$$

Therefore

$$P'_n \widetilde{g} = P'_n \widehat{u} + \frac{1 - P'_n \widehat{u}}{1 + \lambda_n \|1\|_{\mathcal{G}}} = 1 + \lambda_n \frac{\|1\|_{\mathcal{G}}^2}{1 + \lambda_n \|1\|_{\mathcal{G}}^2} (P'_n \widehat{u} - 1). \tag{38}$$

If we can show that $\widehat{u}$ is bounded (i.e., $\mathcal{O}_p(1)$), then $P'_n \widetilde{g} - 1 = \mathcal{O}_p(\lambda_m)$. To show that, we bound $\|\widetilde{g}\|$ because

$$\|\widehat{u}\|_\infty \leq \|\widehat{u}\|_{\mathcal{G}} \leq \sqrt{\|\widehat{u}\|_{\mathcal{G}}^2 + \|\widehat{\beta}\|_{\mathcal{G}}^2} = \|\widetilde{g}\|_{\mathcal{G}}.$$

We have

$$\begin{aligned}
\|\widetilde{g}\|_{\mathcal{G}}^2 &= \langle \widehat{h}_2, (\widehat{T}_{p'} + \lambda_n)^{-2} \widehat{h}_2 \rangle \\
&= \langle (T_{p'} + \lambda_n)^{-1} \widehat{h}_2, [(T_{p'} + \lambda_n)(\widehat{T}_{p'} + \lambda_n)^{-2}(T_{p'} + \lambda_n)](T_{p'} + \lambda_n)^{-1} \widehat{h}_2 \rangle
\end{aligned}$$

Here

$$(T_{p'} + \lambda_n)(\widehat{T}_{p'} + \lambda_n)^{-1} = (I - (T_{p'} - \widehat{T}_{p'})(T_{p'} + \lambda_n)^{-1})^{-1} \tag{39}$$

and on the event $\mathcal{A}_1$ with the condition Eq.(13), we have

$$\|(T_{p'} - \widehat{T}_{p'})(T_{p'} + \lambda_n)^{-1}\|_{\mathcal{L}(\mathcal{G})} \leq \frac{1}{2}.$$

Hence

$$\|(T_{p'}+\lambda_n)(\widehat{T}_{p'}+\lambda_n)^{-1}\|_{\mathcal{L}(\mathcal{G})} \le 2 \tag{40}$$

on the event $\mathcal{A}_1$ with the condition Eq.(13).

We have that

$$\begin{aligned}\|(T_{p'}+\lambda_n)^{-1}(\widehat{h}_2-h_2)\|_{\mathcal{G}} &\le \lambda_n^{-\frac{1}{2}}\|(T_{p'}+\lambda_n)^{-\frac{1}{2}}(\widehat{h}_2-h_2)\|_{\mathcal{G}}\\ &\le 2\log(12/\eta)\left(\frac{2}{\lambda_n n}+\sqrt{\frac{\mathcal{N}(\lambda_n)\|g^*\|_{\mathcal{G}}}{\lambda_n n}}\right),\end{aligned} \tag{41}$$

on the event $\mathcal{A}_3$. Hence Eqs.(40) and (41) and

$$\|(T_{p'}+\lambda_n)^{-1}h_2\| = \|(T_{p'}+\lambda_n)^{-1}T_{p'}g^*\| \le \|g^*\|_{\mathcal{G}}$$

give

$$\|\widetilde{g}\|^2 \le 8\left(\|g^*\|_{\mathcal{G}}^2 + 8\log^2(12/\eta)\left(\frac{4}{\lambda_n^2 n^2}+\frac{\mathcal{N}(\lambda_n)\|g^*\|_{\mathcal{G}}}{\lambda_n n}\right)\right), \tag{42}$$

on the event $\mathcal{A}_3$.

Therefore, Eqs.(38) and (42) give

$$\begin{aligned}|P_n'\widetilde{g}-1| &= \lambda_n\frac{\|1\|_{\mathcal{G}}^2}{1+\lambda_n\|1\|_{\mathcal{G}}^2}|P_n'\widehat{u}-1|\\ &\le \lambda_n\frac{\|1\|_{\mathcal{G}}^2}{1+\lambda_n\|1\|_{\mathcal{G}}^2}\left\{8\left(\|g^*\|_{\mathcal{G}}^2+8\log^2(12/\eta)\left(\frac{4}{\lambda_n^2n^2}+\frac{\mathcal{N}(\lambda_n)\|g^*\|_{\mathcal{G}}}{\lambda_n n}\right)\right)+1\right\}\\ &=: \lambda_n C_{n,\eta},\end{aligned} \tag{43}$$

on the event $\mathcal{A}_3$.

Step 4. Bounding $P'(\widetilde{g}-g^*)^2$

Decompose $\widetilde{g}-g^*$ as

$$\widetilde{g}-g^* = (\widetilde{g}-g_{\lambda_n}) + (g_{\lambda_n}-g^*).$$

The first term is evaluated as follows:

$$\begin{aligned}\widetilde{g}-g_{\lambda_n} &= (\widehat{T}_{p'}+\lambda_n)^{-1}\widehat{h}_2 - (T_{p'}+\lambda_n)^{-1}h_2\\ &= (\widehat{T}_{p'}+\lambda_n)^{-1}\left\{(\widehat{h}_2-h_2)+(T_{p'}-\widehat{T}_{p'})(T_{p'}+\lambda_n)^{-1}h_2\right\}.\end{aligned} \tag{44}$$

Thus

$$\begin{aligned}P'(\widetilde{g}-g_{\lambda_n})^2 &= \left\|\sqrt{T_{p'}}(\widehat{T}_{p'}+\lambda_n)^{-1}\left\{(\widehat{h}_2-h_2)+(T_{p'}-\widehat{T}_{p'})(T_{p'}+\lambda_n)^{-1}h_2\right\}\right\|_{\mathcal{G}}^2\\ &\leq 2\left\{\underbrace{\|\sqrt{T_{p'}}(\widehat{T}_{p'}+\lambda_n)^{-1}(\widehat{h}_2-h_2)\|_{\mathcal{G}}^2}_{\text{(4-a)}}\right.\\ &\qquad\left.+\underbrace{\|\sqrt{T_{p'}}(\widehat{T}_{p'}+\lambda_n)^{-1}(T_{p'}-\widehat{T}_{p'})(T_{p'}+\lambda_n)^{-1}h_2\|_{\mathcal{G}}^2}_{\text{(4-b)}}\right\}. \qquad (45)\end{aligned}$$

Bounding (4-a): We have

$$\begin{aligned}&\|\sqrt{T_{p'}}(\widehat{T}_{p'}+\lambda_n)^{-1}(\widehat{h}_2-h_2)\|_{\mathcal{G}}\\ &\leq \|\sqrt{T_{p'}}(T_{p'}+\lambda_n)^{-\frac{1}{2}}\|_{\mathcal{L}(\mathcal{G})}\|(T_{p'}+\lambda_n)^{\frac{1}{2}}(\widehat{T}_{p'}+\lambda_n)^{-1}(T_{p'}+\lambda_n)^{\frac{1}{2}}\|_{\mathcal{G}}\\ &\quad\times\|(T_{p'}+\lambda_n)^{-\frac{1}{2}}(\widehat{h}_2-h_2)\|_{\mathcal{G}}.\end{aligned}$$

It is obvious that

$$\|\sqrt{T_{p'}}(T_{p'}+\lambda_n)^{-\frac{1}{2}}\|_{\mathcal{G}}\leq 1. \qquad (46)$$

By Eq.(24),

$$\|(T_{p'}+\lambda_n)^{\frac{1}{2}}(\widehat{T}_{p'}+\lambda_n)^{-1}(T_{p'}+\lambda_n)^{\frac{1}{2}}\|_{\mathcal{G}}\leq 2 \qquad (47)$$

on the event $\mathcal{A}_1$. By Eq.(28),

$$\|(T_{p'}+\lambda_n)^{-\frac{1}{2}}(\widehat{h}_2-h_2)\|_{\mathcal{G}}\leq 2\log(12/\eta)\left(\frac{2}{n\sqrt{\lambda_n}}+\sqrt{\frac{\|g^*\|_{\mathcal{G}}\mathcal{N}(\lambda_n)}{n}}\right), \qquad (48)$$

on the event $\mathcal{A}_3$.

Combining Eqs.(46), (24), and (28), we have

$$\|\sqrt{T_{p'}}(\widehat{T}_{p'}+\lambda_n)^{-1}(\widehat{h}_2-h_2)\|_{\mathcal{G}}\leq 4\log(12/\eta)\left(\frac{2}{n\sqrt{\lambda_n}}+\sqrt{\frac{\|g^*\|_{\mathcal{G}}\mathcal{N}(\lambda_n)}{n}}\right), \qquad (49)$$

on the event $\mathcal{A}_1\cap\mathcal{A}_3$.

Bounding (4-b): We have

$$\begin{aligned}&\|\sqrt{T_{p'}}(\widehat{T}_{p'}+\lambda_n)^{-1}(T_{p'}-\widehat{T}_{p'})(T_{p'}+\lambda_n)^{-1}h_2\|_{\mathcal{G}}\\ &\leq \|\sqrt{T_{p'}}(T_{p'}+\lambda_n)^{-\frac{1}{2}}\|_{\mathcal{L}(G)}\|(T_{p'}+\lambda_n)^{\frac{1}{2}}(\widehat{T}_{p'}+\lambda_n)^{-1}(T_{p'}+\lambda_n)^{\frac{1}{2}}\|_{\mathcal{L}(G)}\\ &\quad\times\|(T_{p'}+\lambda_n)^{-\frac{1}{2}}(T_{p'}-\widehat{T}_{p'})(T_{p'}+\lambda_n)^{-1}h_2\|_{\mathcal{G}}.\end{aligned}$$

By Eq.(33), we have

$$\|(T_{p'}+\lambda_n)^{-\frac{1}{2}}(T_{p'}-\widehat{T}_{p'})(T_{p'}+\lambda_n)^{-1}h_2\|_{\mathcal{G}} \leq 2\log(12/\eta)\left(\frac{2}{n\sqrt{\lambda_n}}+\sqrt{\frac{\|g^*\|_{\mathcal{G}}^2\mathcal{N}(\lambda_n)}{n}}\right),$$

on the event $\mathcal{A}_4$. Thus Eqs.(46), (24), and (33) indicate

$$\begin{aligned}&\|\sqrt{T_{p'}}(\widehat{T}_{p'}+\lambda_n)^{-1}(T_{p'}-\widehat{T}_{p'})(T_{p'}+\lambda_n)^{-1}h_2\|_{\mathcal{G}}\\ &\leq 4\log(12/\eta)\left(\frac{2}{n\sqrt{\lambda_n}}+\sqrt{\frac{\|g^*\|_{\mathcal{G}}^2\mathcal{N}(\lambda_n)}{n}}\right),\end{aligned} \tag{50}$$

on the event $\mathcal{A}_1\cap\mathcal{A}_4$.

Combining the bounds of (4-a) and (4-b): Substituting Eqs.(50) and (49) to Eq.(45), we have

$$\begin{aligned}P'(\widetilde{g}-g_{\lambda_n})^2 \leq & 64\left\{\log^2(12/\eta)\left(\frac{4}{n^2\lambda_n}+\frac{\|g^*\|_{\mathcal{G}}^2\mathcal{N}(\lambda_n)}{n}\right)\right.\\ &\left.+\log^2(12/\eta)\left(\frac{4}{n^2\lambda_n}+\frac{\|g^*\|_{\mathcal{G}}\mathcal{N}(\lambda_n)}{n}\right)\right\},\end{aligned} \tag{51}$$

on the event $\mathcal{A}_1\cap\mathcal{A}_4$.

On the other hand, $P'(g_{\lambda_n}-g^*)^2$ is bounded as

$$\begin{aligned}P'(g_{\lambda_n}-g^*)^2 &= \|\sqrt{T_{p'}}((T_{p'}+\lambda_n)^{-1}h_2-g^*)\|_{\mathcal{G}}^2 = \|\sqrt{T_{p'}}(T_{p'}+\lambda_n)^{-1}(h_2-(T_{p'}+\lambda_n)g^*)\|_{\mathcal{G}}^2\\ &= \|\sqrt{T_{p'}}(T_{p'}+\lambda_n)^{-1}\lambda_n g^*)\|_{\mathcal{G}}^2 \leq \|(T_{p'}+\lambda_n)^{-\frac{1}{2}}\lambda_n g^*)\|_{\mathcal{G}}^2 \leq \lambda_n\|g^*\|_{\mathcal{G}}^2.\end{aligned} \tag{52}$$

By Eqs.(51) and (52), $P'(\widetilde{g}-g^*)^2$ is bounded as

$$\begin{aligned}&P'(\widetilde{g}-g^*)^2\\ &\leq 2(P'(\widetilde{g}-g_{\lambda_n})^2+P'(g_{\lambda_n}-g^*)^2)\\ &\leq 128\log^2(12/\eta)\left(\frac{8}{n^2\lambda_n}+\frac{(\|g^*\|_{\mathcal{G}}+\|g^*\|_{\mathcal{G}}^2)\mathcal{N}(\lambda_n)}{n}\right)+2\lambda_n\|g^*\|_{\mathcal{G}}^2,\end{aligned}$$

on the event $\mathcal{A}_1\cap\mathcal{A}_4$.

Step 5. Bounding $|(P_n'-P')(g^*-1)|$ and $|(P_n-P)(g^*-1)|$

By Proposition 1, we have the following bound

$$|(P_n'-P')(g^*-1)| \leq 2\log(12/\eta)\left(\frac{2\|g^*-1\|_\infty}{n}+\sqrt{\frac{P'(g^*-1)^2}{n}}\right), \tag{53}$$

with probability $1-\eta/6$. Similarly we have

$$|(P_n-P)(g^*-1)| \le 2\log(12/\eta)\left(\frac{2\|g^*-1\|_\infty}{n}+\sqrt{\frac{P(g^*-1)^2}{n}}\right), \tag{54}$$

with probability $1-\eta/6$.

We define $\mathcal{A}_5$ and $\mathcal{A}_6$ as the events where the above inequalities hols:

$$\mathcal{A}_5 := \left\{|(P'_n-P')(g^*-1)| \le 2\log(12/\eta)\left(\frac{2\|g^*-1\|_\infty}{n}+\sqrt{\frac{P'(g^*-1)^2}{n}}\right)\right\}, \tag{55}$$

$$\mathcal{A}_6 := \left\{|(P_n-P)(g^*-1)| \le 2\log(12/\eta)\left(\frac{2\|g^*-1\|_\infty}{n}+\sqrt{\frac{P(g^*-1)^2}{n}}\right)\right\}. \tag{56}$$

Step 6. Combining the bounds of Step 1 to 5.

Finally we obtain

$$\begin{aligned}
&|\widehat{\mathrm{PE}}(\mathcal{X},\mathcal{X}')-\mathrm{PE}(P,P')| \\
&\le 8\log(12/\eta)^2\left(\frac{8}{n^2\lambda_n}+\frac{(\|g^*\|_{\mathcal{G}}+\sqrt{\|g^*\|_{\mathcal{G}}})\mathcal{N}(\lambda_n)}{n}\right) \\
&\quad+\log(12/\eta)\left(\frac{2\|g^*\|_{\mathcal{G}}}{n}+\|g^*\|_{\mathcal{G}}\sqrt{\frac{\lambda_n\mathcal{N}(\lambda_n)}{n}}\right) \\
&\quad+8\log(12/\eta)^2\left(\frac{8}{n^2\lambda_n}+\frac{(\|g^*\|_{\mathcal{G}}^{\frac{3}{2}}+\|g^*\|_{\mathcal{G}})\mathcal{N}(\lambda_n)}{n}\right) \\
&\quad+\log(12/\eta)\left(\frac{2\|g^*\|_{\mathcal{G}}}{n}+\|g^*\|_{\mathcal{G}}^{\frac{3}{2}}\sqrt{\frac{\lambda_n\mathcal{N}(\lambda_n)}{n}}\right) \\
&\quad+\frac{3}{2}\lambda_n C_{n,\eta} \\
&\quad+\log(12/\eta)\left(\frac{4\|g^*-1\|_\infty}{n}+\sqrt{\frac{P'(g^*-1)^2}{n}}+\sqrt{\frac{P(g^*-1)^2}{n}}\right) \\
&\quad+\frac{1}{2}\sqrt{P'(g^*-1)^2}\sqrt{128\log^2(12/\eta)\left(\frac{8}{n^2\lambda_n}+\frac{(\|g^*\|_{\mathcal{G}}+\|g^*\|_{\mathcal{G}}^2)\mathcal{N}(\lambda_n)}{n}\right)+2\lambda_n\|g^*\|_{\mathcal{G}}^2},
\end{aligned}$$

on the event $\bigcap_{\ell=1}^{6}\mathcal{A}_\ell$ the probability of which is at least $1-\eta$. □

Proof of Theorem 1'. By Proposition 3 in Caponnetto and de Vito (2007), we obtain

$$\mathcal{N}(\lambda) \le \frac{2c}{2-\gamma}\lambda^{-\frac{\gamma}{2}},$$

where c is the constant appears in the assumption. Then substituting the above inequality and $\lambda_n = \left(\frac{\log n}{n}\right)^{\frac{2}{2+\gamma}}$ to Eq.(14), we can see that there is a constant K depending on $c, \gamma, \|g\|_{\mathcal{G}}$ such that

$$\begin{aligned}
&|\widehat{\mathrm{PE}}(\mathcal{X}, \mathcal{X}') - \mathrm{PE}(P, P')| \\
&\leq K \Biggl\{ \left(\log(12/\eta)^2 + \log(12/\eta) + 1\right) \left(\frac{\log n}{n}\right)^{\frac{2}{2+\gamma}} \\
&\qquad + \log(12/\eta) \left(\sqrt{\frac{P'(g^* - 1)^2}{n}} + \frac{\|g^* - 1\|_\infty}{n} \right) \\
&\qquad + \sqrt{P'(g^* - 1)^2} \sqrt{(\log(12/\eta)^2 + 1)} \left(\frac{\log n}{n}\right)^{\frac{1}{2+\gamma}} \Biggr\},
\end{aligned} \tag{57}$$

with probability at least $1 - \eta$ under the condition Eq.(13). The condition Eq.(13) is satisfied for sufficiently large n. Therefore Eq.(57) implies that

$$|\widehat{\mathrm{PE}}(\mathcal{X}, \mathcal{X}') - \mathrm{PE}(P, P')| = \mathcal{O}_p \left(\left(\frac{\log n}{n}\right)^{\frac{2}{2+\gamma}} + \sqrt{P'(g^* - 1)^2} \left(\frac{\log n}{n}\right)^{\frac{1}{2+\gamma}} \right).$$

□

B Proof of Theorem 2

In this section, we prove Theorem 2. Here, for being more precise, we rewrite Theorem 2 as follow.

Theorem 2'. *Let* $\widetilde{F}_n(\cdot|\mathcal{X} \cup \mathcal{X}')$ *be the distribution function of* $\widehat{\mathrm{PE}}(\widetilde{\mathcal{X}}, \widetilde{\mathcal{X}}')$ *given* $\mathcal{X} \cup \mathcal{X}'$. *Let*

$$\widetilde{q}(\mathcal{X} \cup \mathcal{X}') = \sup\{x \in \mathbb{R} \mid \widetilde{F}_n(x|\mathcal{X} \cup \mathcal{X}') \leq 1 - \alpha\}$$

be the upper 100α*-percentile point. Then, if the null hypothesis is true (i.e.,* $P = P'$*),*

$$\mathrm{Prob}\left(\widehat{\mathrm{PE}}(\mathcal{X}, \mathcal{X}') > \widetilde{q}(\mathcal{X} \cup \mathcal{X}')\right) \leq \alpha.$$

Proof. Since the samples $\{\boldsymbol{x}_i\}_{i=1}^n$ and $\{\boldsymbol{x}'_i\}_{i=1}^n$ are distributed i.i.d. and $P = P'$, they are *exchangeable*, i.e., the distribution of $(\boldsymbol{y}_1, \ldots, \boldsymbol{y}_{2n}) = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n, \boldsymbol{x}'_1, \ldots, \boldsymbol{x}'_n)$ is same as that of $(\boldsymbol{y}_{\tau(1)}, \ldots, \boldsymbol{y}_{\tau(2n)})$ for any permutation τ on $\{1, \ldots, 2n\}$. This means that the distribution function $\widetilde{F}_n(\cdot \mid \mathcal{S})$ is the same as that of $\mathrm{PE}(\mathcal{X}, \mathcal{X}')$ conditioned on $\mathcal{S} = \mathcal{X} \cup \mathcal{X}'$. Then, we have

$$\begin{aligned}
\mathrm{Prob}\left(\widehat{\mathrm{PE}}(\mathcal{X}, \mathcal{X}') > \widetilde{q}(\mathcal{X} \cup \mathcal{X}')\right) &= \mathrm{E}_{\mathcal{X} \cup \mathcal{X}'} \left[\mathrm{Prob}\left(\widehat{\mathrm{PE}}(\mathcal{X}, \mathcal{X}') > \widetilde{q}(\mathcal{X} \cup \mathcal{X}') \mid \mathcal{X} \cup \mathcal{X}'\right)\right] \\
&= \mathrm{E}_{\mathcal{X} \cup \mathcal{X}'} \left[1 - \widetilde{F}_n\left(\widetilde{q}(\mathcal{X} \cup \mathcal{X}') \mid \mathcal{X} \cup \mathcal{X}'\right)\right] \\
&\leq \alpha,
\end{aligned}$$

which concludes the proof. □

References

Ali, S. M., & Silvey, S. D. (1966). A general class of coefficients of divergence of one distribution from another. *Journal of the Royal Statistical Society, Series B*, *28*, 131–142.

Anderson, N., Hall, P., & Titterington, D. (1994). Two-sample test statistics for measuring discrepancies between two multivariate probability density functions using kernel-based density estimates. *Journal of Multivariate Analysis*, *50*, 41–54.

Aronszajn, N. (1950). Theory of reproducing kernels. *Transactions of the American Mathematical Society*, *68*, 337–404.

Bachman, G., & Narici, L. (2000). *Functional analysis*. Mineola, NY, USA: Dover Publications.

Biau, G., & Györfi, L. (2005). On the asymptotic properties of a nonparametric l1-test statistic of homogeneity. *IEEE Transactions on Information Theory*, *51*, 3965–3973.

Bickel, P. (1969). A distribution free version of the Smirnov two sample test in the p-variate case. *The Annals of Mathematical Statistics*, *40*, 1–23.

Borgwardt, K. M., Gretton, A., Rasch, M. J., Kriegel, H.-P., Schölkopf, B., & Smola, A. J. (2006). Integrating structured biological data by kernel maximum mean discrepancy. *Bioinformatics*, *22*, e49–e57.

Caponnetto, A., & de Vito, E. (2007). Optimal rates for regularized least-squares algorithm. *Foundations of Computational Mathematics*, *7*, 331–368.

Caruana, R., Pratt, L., & Thrun, S. (1997). Multitask learning. *Machine Learning*, *28*, 41–75.

Cheng, K. F., & Chu, C. K. (2004). Semiparametric density estimation under a two-sample density ratio model. *Bernoulli*, *10*, 583–604.

Collins, M., & Duffy, N. (2002). Convolution kernels for natural language. *Advances in Neural Information Processing Systems 14* (pp. 625–632). Cambridge, MA: MIT Press.

Csiszár, I. (1967). Information-type measures of difference of probability distributions and indirect observation. *Studia Scientiarum Mathematicarum Hungarica*, *2*, 229–318.

Darbellay, G. A., & Vajda, I. (1999). Estimation of the information by an adaptive partitioning of the observation space. *IEEE Transactions on Information Theory*, *45*, 1315–1321.

Duffy, N., & Collins, M. (2002). Convolution kernels for natural language. *Advances in Neural Information Processing Systems 14* (pp. 625–632). Cambridge, MA: MIT Press.

Efron, B., & Tibshirani, R. J. (1993). *An introduction to the bootstrap.* New York: Chapman & Hall.

Friedman, J., & Rafsky, L. (1979). Multivariate generalizations of the Wald-Wolfowitz and Smirnov two-sample tests. *The Annals of Statistics*, *7*, 697–717.

Gärtner, T. (2003). A survey of kernels for structured data. *SIGKDD Explorations*, *5*, S268–S275.

Gärtner, T., Flach, P., & Wrobel, S. (2003). On graph kernels: Hardness results and efficient alternatives. *Proceedings of the Sixteenth Annual Conference on Computational Learning Theory* (pp. 129–143).

Gretton, A., Borgwardt, K. M., M.Rasch, Schölkopf, B., & Smola, A. J. (2007). A kernel method for the two-sample-problem. In B. Schölkopf, J. Platt and T. Hoffman (Eds.), *Advances in neural information processing systems 19*, 513–520. Cambridge, MA: MIT Press.

Gretton, A., Smola, A., Huang, J., Schmittfull, M., Borgwardt, K., & Schölkopf, B. (2009). Covariate shift by kernel mean matching. In J. Quiñonero-Candela, M. Sugiyama, A. Schwaighofer and N. Lawrence (Eds.), *Dataset shift in machine learning*, chapter 8, 131–160. Cambridge, MA: MIT Press.

Hachiya, H., Akiyama, T., Sugiyama, M., & Peters, J. (2009). Adaptive importance sampling for value function approximation in off-policy reinforcement learning. *Neural Networks*, *22*, 1399–1410.

Hall, P., & Tajvidi, N. (2002). Permutation tests for equality of distributions in highdimensional settings. *Biometrika*, *89*, 359–374.

Hastie, T., Tibshirani, R., & Friedman, J. (2001). *The elements of statistical learning: Data mining, inference, and prediction.* New York: Springer.

Hotelling, H. (1951). A generalized t test and measure of multivariate dispersion. *Proceedings of the 2nd Berkeley Symposium on Mathematical Statistics and Probability* (pp. 23–41). Berkeley, CA., USA: University of California Press.

Huang, J., Smola, A., Gretton, A., Borgwardt, K. M., & Schölkopf, B. (2007). Correcting sample selection bias by unlabeled data. In B. Schölkopf, J. Platt and T. Hoffman (Eds.), *Advances in neural information processing systems 19*, 601–608. Cambridge, MA, USA: MIT Press.

Kanamori, T., Hido, S., & Sugiyama, M. (2009a). A least-squares approach to direct importance estimation. *Journal of Machine Learning Research*, *10*, 1391–1445.

Kanamori, T., Suzuki, T., & Sugiyama, M. (2009b). *Condition number analysis of kernel-based density ratio estimation* (Technical Report). arXiv.

Kanamori, T., Suzuki, T., & Sugiyama, M. (2010). Theoretical analysis of density ratio estimation. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, *E93-A*, 787–798.

Kashima, H., & Koyanagi, T. (2002). Kernels for semi-structured data. *Proceedings of the Nineteenth International Conference on Machine Learning* (pp. 291–298). San Francisco, CA: Morgan Kaufmann.

Kashima, H., Tsuda, K., & Inokuchi, A. (2003). Marginalized kernels between labeled graphs. *Proceedings of the Twentieth International Conference on Machine Learning* (pp. 321–328). San Francisco, CA: Morgan Kaufmann.

Keziou, A., & Leoni-Aubin, S. (2005). Test of homogeneity in semiparametric two-sample density ratio models. *Comptes Rendus Mathématique*, *340*, 905–910.

Kondor, R. I., & Lafferty, J. (2002). Diffusion kernels on graphs and other discrete input spaces. *Proceedings of the Nineteenth International Conference on Machine Learning* (pp. 315–322).

Kullback, S. (1959). *Information theory and statistics*. New York: Wiley.

Kullback, S., & Leibler, R. A. (1951). On information and sufficiency. *Annals of Mathematical Statistics*, *22*, 79–86.

Li, Q. (1996). Nonparametric testing of closeness between two unknown distribution functions. *Econometric Reviews*, *15*, 261–274.

Lodhi, H., Saunders, C., Shawe-Taylor, J., Cristianini, N., & Watkins, C. (2002). Text classification using string kernels. *Journal of Machine Learning Research*, *2*, 419–444.

Müller, A. (1997). Integral probability metrics and their generating classes of functions. *Advances in Applied Probability*, *29*, 429–443.

Nguyen, X., Wainwright, M. J., & Jordan, M. I. (2010). Estimating divergence functionals and the likelihood ratio by convex risk minimization. *IEEE Transactions on Information Theory*. to appear.

Pearson, K. (1900). On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling. *Philosophical Magazine*, *50*, 157–175.

Pérez-Cruz, F. (2008). Kullback-Leibler divergence estimation of continuous distributions. *Proceedings of IEEE International Symposium on Information Theory* (pp. 1666–1670). Nice, France.

Qin, J. (1998). Inferences for case-control and semiparametric two-sample density ratio models. *Biometrika*, *85*, 619–639.

Rätsch, G., Onoda, T., & Müller, K.-R. (2001). Soft margins for adaboost. *Machine Learning*, *42*, 287–320.

Schölkopf, B., & Smola, A. J. (2002). *Learning with kernels*. Cambridge, MA: MIT Press.

Silva, J., & Narayanan, S. (2007). Universal consistency of data-driven partitions for divergence estimation. *Proceedings of IEEE International Symposium on Information Theory* (pp. 2021–2025). Nice, France.

Sriperumbudur, B., Fukumizu, K., Gretton, A., Lanckriet, G., & Schölkopf, B. (2009). Kernel choice and classifiability for rkhs embeddings of probability distributions. In Y. Bengio, D. Schuurmans, J. Lafferty, C. K. I. Williams and A. Culotta (Eds.), *Advances in neural information processing systems 22*, 1750–1758. Cambridge, MA: MIT Press.

Steinwart, I. (2001). On the influence of the kernel on the consistency of support vector machines. *Journal of Machine Learning Research*, *2*, 67–93.

Student (1908). The probable error of a mean. *Biometrika*, *6*, 1–25.

Sugiyama, M., Kawanabe, M., & Chui, P. L. (2010). Dimensionality reduction for density ratio estimation in high-dimensional spaces. *Neural Networks*, *23*, 44–59.

Sugiyama, M., Krauledat, M., & Müller, K.-R. (2007). Covariate shift adaptation by importance weighted cross validation. *Journal of Machine Learning Research*, *8*, 985–1005.

Sugiyama, M., Suzuki, T., Nakajima, S., Kashima, H., von Bünau, P., & Kawanabe, M. (2008). Direct importance estimation for covariate shift adaptation. *Annals of the Institute of Statistical Mathematics*, *60*, 699–746.

Sugiyama, M., Yamada, M., von Bünau, P., Suzuki, T., Kanamori, T., & Kawanabe, M. (2011). Direct density-ratio estimation with dimensionality reduction via least-squares hetero-distributional subspace search. *Neural Networks*.

Wang, Q., Kulkarmi, S. R., & Verdú, S. (2005). Divergence estimation of contiunous distributions based on data-dependent partitions. *IEEE Transactions on Information Theory*, *51*, 3064–3074.

Yamada, M., Sugiyama, M., Wichern, G., & Simm, J. (2010). Direct importance estimation with a mixture of probabilistic principal component analyzers. *IEICE Transactions on Information and Systems*, *E93-D*, 2846–2849.

Density-Ratio Matching under the Bregman Divergence: A Unified Framework of Density-Ratio Estimation

Masashi Sugiyama
Tokyo Institute of Technology, Japan.
sugi@cs.titech.ac.jp
http://sugiyama-www.cs.titech.ac.jp/˜sugi

Taiji Suzuki
The University of Tokyo, Japan.
s-taiji@stat.t.u-tokyo.ac.jp

Takafumi Kanamori
Nagoya University, Japan.
kanamori@is.nagoya-u.ac.jp

Abstract

Estimation of the ratio of probability densities has attracted a great deal of attention since it can be used for addressing various statistical paradigms. A naive approach to density-ratio approximation is to first estimate numerator and denominator densities separately and then take their ratio. However, this two-step approach does not perform well in practice, and methods for directly estimating density ratios without density estimation have been explored. In this paper, we first give a comprehensive review of existing density-ratio estimation methods and discuss their pros and cons. Then we propose a new framework of density-ratio estimation in which a density-ratio model is fitted to the true density-ratio under the Bregman divergence. Our new framework includes existing approaches as special cases, and is substantially more general. Finally, we develop a robust density-ratio estimation method under the power divergence, which is a novel instance in our framework.

Keywords

Density ratio, Bregman divergence, Logistic regression, Kernel mean matching, Kullback-Leibler importance estimation procedure, Least-squares importance fitting

1 Introduction

The *ratio* of probability densities can be used for various statistical data processing purposes (Sugiyama et al., 2009, 2012) such as discriminant analysis (Silverman, 1978), nonstationarity adaptation (Shimodaira, 2000; Sugiyama and Müller, 2005; Sugiyama et al., 2007; Quiñonero-Candela et al., 2009; Sugiyama and Kawanabe, 2011), multi-task learning (Bickel et al., 2008), outlier detection (Hido et al., 2008; Smola et al., 2009; Hido et al., 2011), two-sample test (Keziou and Leoni-Aubin, 2005; Sugiyama et al., 2011a) change detection in time series (Kawahara and Sugiyama, 2009), conditional density estimation (Sugiyama et al., 2010), and probabilistic classification (Sugiyama, 2010).

Furthermore, *mutual information*—which plays a central role in information theory (Cover and Thomas, 2006)—can be estimated via density-ratio estimation (Suzuki et al., 2008, 2009b). Since mutual information is a measure of statistical independence between random variables, density-ratio estimation can be used also for variable selection (Suzuki et al., 2009a), dimensionality reduction (Suzuki and Sugiyama, 2010), independent component analysis (Suzuki and Sugiyama, 2009), causal inference (Yamada and Sugiyama, 2010), clustering (Kimura and Sugiyama, 2011), and cross-domain object matching (Yamada and Sugiyama, 2011) Thus, density-ratio estimation is a versatile tool for statistical data processing.

A naive approach to approximating a density-ratio is to separately estimate the two densities corresponding to the numerator and denominator of the ratio, and then take the ratio of the estimated densities. However, this naive approach is not reliable in high-dimensional problems since division by an estimated quantity can magnify the estimation error of the dividend. To overcome this drawback, various approaches to directly estimating density-ratios without going through density estimation have been explored recently, including the *moment matching approach* (Gretton et al., 2009), the *probabilistic classification approach* (Qin, 1998; Cheng and Chu, 2004), the *density matching approach* (Sugiyama et al., 2008; Tsuboi et al., 2009; Yamada and Sugiyama, 2009; Nguyen et al., 2010; Yamada et al., 2010), and the *density-ratio fitting approach* (Kanamori et al., 2009).

The purpose of this paper is to provide a general framework of density-ratio estimation that accommodates the above methods. More specifically, we propose a new density-ratio estimation approach called *density-ratio matching*—a density-ratio model is fitted to the true density-ratio function under the *Bregman divergence* (Bregman, 1967). We further develop a robust density-ratio estimation method under the *power divergence* (Basu et al., 1998), which is a novel instance in our general framework. Note that the Bregman divergence has been widely used in machine learning literature so far (Collins et al., 2002; Murata et al., 2004; Tsuda et al., 2005; Dhillon and Sra, 2006; Cayton, 2008; Wu et al., 2009), and the current paper explores a new application of the Bregman divergence in the framework of density-ratio estimation.

The rest of this paper is organized as follows. After the problem formulation below, we give a comprehensive review of density-ratio estimation methods in Section 2. In Section 3, we describe our new framework for density-ratio estimation. Finally, we conclude in Section 4.

Problem Formulation: The problem of density-ratio estimation addressed in this paper is formulated as follows. Let $\mathcal{X}$ $(\subset \mathbb{R}^d)$ be the data domain, and suppose we are given independent and identically distributed (i.i.d.) samples $\{\boldsymbol{x}_i^{\mathrm{nu}}\}_{i=1}^{n_{\mathrm{nu}}}$ from a distribution with density $p_{\mathrm{nu}}^*(\boldsymbol{x})$ defined on $\mathcal{X}$ and i.i.d. samples $\{\boldsymbol{x}_j^{\mathrm{de}}\}_{j=1}^{n_{\mathrm{de}}}$ from another distribution with density $p_{\mathrm{de}}^*(\boldsymbol{x})$ defined on $\mathcal{X}$.

$$\{\boldsymbol{x}_i^{\mathrm{nu}}\}_{i=1}^{n_{\mathrm{nu}}} \overset{\text{i.i.d.}}{\sim} p_{\mathrm{nu}}^*(\boldsymbol{x}) \quad \text{and} \quad \{\boldsymbol{x}_j^{\mathrm{de}}\}_{j=1}^{n_{\mathrm{de}}} \overset{\text{i.i.d.}}{\sim} p_{\mathrm{de}}^*(\boldsymbol{x}).$$

We assume that $p_{\mathrm{de}}^*(\boldsymbol{x})$ is strictly positive over the domain $\mathcal{X}$. The goal is to estimate the density-ratio,

$$r^*(\boldsymbol{x}) := \frac{p_{\mathrm{nu}}^*(\boldsymbol{x})}{p_{\mathrm{de}}^*(\boldsymbol{x})},$$

from samples $\{\boldsymbol{x}_i^{\mathrm{nu}}\}_{i=1}^{n_{\mathrm{nu}}}$ and $\{\boldsymbol{x}_j^{\mathrm{de}}\}_{j=1}^{n_{\mathrm{de}}}$. 'nu' and 'de' indicate 'numerator' and 'denominator', respectively.

2 Existing Density-Ratio Estimation Methods

In this section, we give a comprehensive review of existing density-ratio estimation methods.

2.1 Moment Matching

Here, we describe a framework of density-ratio estimation based on *moment matching*.

2.1.1 Finite-Order Approach

First, we describe methods of finite-oder moment-matching for density-ratio estimation.

The simplest implementation of moment matching would be to match the first-order moment (i.e., the mean):

$$\underset{r}{\operatorname{argmin}} \left\| \int \boldsymbol{x} r(\boldsymbol{x}) p_{\mathrm{de}}^*(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} - \int \boldsymbol{x} p_{\mathrm{nu}}^*(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \right\|^2,$$

where $\|\cdot\|$ denotes the Euclidean norm. Its non-linear variant can be obtained using some non-linear function $\boldsymbol{\phi}(\boldsymbol{x}) : \mathbb{R}^d \to \mathbb{R}^t$ as

$$\underset{r}{\operatorname{argmin}}\ \mathrm{MM}'(r),$$

where

$$\mathrm{MM}'(r) := \left\| \int \boldsymbol{\phi}(\boldsymbol{x}) r(\boldsymbol{x}) p_{\mathrm{de}}^*(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} - \int \boldsymbol{\phi}(\boldsymbol{x}) p_{\mathrm{nu}}^*(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \right\|^2.$$

'MM' stands for 'moment matching'. Let us ignore the irrelevant constant in $\mathrm{MM}'(r)$ and define the rest as $\mathrm{MM}(r)$:

$$\begin{aligned}\mathrm{MM}(r) := &\left\| \int \boldsymbol{\phi}(\boldsymbol{x}) r(\boldsymbol{x}) p^*_{\mathrm{de}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \right\|^2 \\ &- 2 \left\langle \int \boldsymbol{\phi}(\boldsymbol{x}) r(\boldsymbol{x}) p^*_{\mathrm{de}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}, \int \boldsymbol{\phi}(\boldsymbol{x}) p^*_{\mathrm{nu}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \right\rangle, \qquad (1)\end{aligned}$$

where $\langle \cdot, \cdot \rangle$ denotes the inner product.

In practice, the expectations over $p^*_{\mathrm{nu}}(\boldsymbol{x})$ and $p^*_{\mathrm{de}}(\boldsymbol{x})$ in $\mathrm{MM}(r)$ are replaced by sample averages. That is, for an n_{de}-dimensional vector

$$\boldsymbol{r}^*_{\mathrm{de}} := (r^*(\boldsymbol{x}^{\mathrm{de}}_1), \ldots, r^*(\boldsymbol{x}^{\mathrm{de}}_{n_{\mathrm{de}}}))^\top,$$

where $^\top$ denotes the transpose, an estimator $\widehat{\boldsymbol{r}}_{\mathrm{de}}$ of $\boldsymbol{r}^*_{\mathrm{de}}$ can be obtained by solving the following optimization problem.

$$\widehat{\boldsymbol{r}}_{\mathrm{de}} := \underset{\boldsymbol{r} \in \mathbb{R}^{n_{\mathrm{de}}}}{\mathrm{argmin}}\ \widehat{\mathrm{MM}}(\boldsymbol{r}), \qquad (2)$$

where

$$\widehat{\mathrm{MM}}(\boldsymbol{r}) := \frac{1}{n^2_{\mathrm{de}}} \boldsymbol{r}^\top \boldsymbol{\Phi}^\top_{\mathrm{de}} \boldsymbol{\Phi}_{\mathrm{de}} \boldsymbol{r} - \frac{2}{n_{\mathrm{de}} n_{\mathrm{nu}}} \boldsymbol{r}^\top \boldsymbol{\Phi}^\top_{\mathrm{de}} \boldsymbol{\Phi}_{\mathrm{nu}} \boldsymbol{1}_{n_{\mathrm{nu}}}. \qquad (3)$$

$\boldsymbol{1}_n$ denotes the n-dimensional vector with all ones. $\boldsymbol{\Phi}_{\mathrm{nu}}$ and $\boldsymbol{\Phi}_{\mathrm{de}}$ are the $t \times n_{\mathrm{nu}}$ and $t \times n_{\mathrm{de}}$ *design matrices* defined by

$$\boldsymbol{\Phi}_{\mathrm{nu}} := (\boldsymbol{\phi}(\boldsymbol{x}^{\mathrm{nu}}_1), \ldots, \boldsymbol{\phi}(\boldsymbol{x}^{\mathrm{nu}}_{n_{\mathrm{nu}}})) \ \text{ and } \ \boldsymbol{\Phi}_{\mathrm{de}} := (\boldsymbol{\phi}(\boldsymbol{x}^{\mathrm{de}}_1), \ldots, \boldsymbol{\phi}(\boldsymbol{x}^{\mathrm{de}}_{n_{\mathrm{de}}})),$$

respectively. Taking the derivative of the objective function (3) with respect to $\boldsymbol{r}$ and setting it to zero, we have

$$\frac{2}{n^2_{\mathrm{de}}} \boldsymbol{\Phi}^\top_{\mathrm{de}} \boldsymbol{\Phi}_{\mathrm{de}} \boldsymbol{r} - \frac{2}{n_{\mathrm{de}} n_{\mathrm{nu}}} \boldsymbol{\Phi}^\top_{\mathrm{de}} \boldsymbol{\Phi}_{\mathrm{nu}} \boldsymbol{1}_{n_{\mathrm{nu}}} = \boldsymbol{0}_t,$$

where $\boldsymbol{0}_t$ denotes the t-dimensional vector with all zeros. Solving this equation with respect to $\boldsymbol{r}$, one can obtain the solution analytically as

$$\widehat{\boldsymbol{r}}_{\mathrm{de}} = \frac{n_{\mathrm{de}}}{n_{\mathrm{nu}}} (\boldsymbol{\Phi}^\top_{\mathrm{de}} \boldsymbol{\Phi}_{\mathrm{de}})^{-1} \boldsymbol{\Phi}^\top_{\mathrm{de}} \boldsymbol{\Phi}_{\mathrm{nu}} \boldsymbol{1}_{n_{\mathrm{nu}}}.$$

One may add a normalization constraint

$$\frac{1}{n_{\mathrm{de}}} \boldsymbol{1}^\top_{n_{\mathrm{de}}} \boldsymbol{r} = 1$$

to the optimization problem (2). Then the optimization problem becomes a *convex linearly-constrained quadratic program.* Since there is no known method for obtaining the analytic-form solution for convex linearly-constrained quadratic programs, a numerical solver may be needed to compute the solution. Furthermore, a non-negativity constraint

$$\boldsymbol{r} \geq \boldsymbol{0}_{n_{\mathrm{de}}}$$

and/or an upper bound for a positive constant B, i.e.,

$$\boldsymbol{r} \leq B\boldsymbol{1}_{n_{\mathrm{de}}}$$

may also be incorporated in the optimization problem (2), where inequalities for vectors are applied in the element-wise manner. Even with these modifications, the optimization problem is still a convex linearly-constrained quadratic program, so its solution can be numerically computed by standard optimization software.

The above *fixed-design* method gives estimates of the density-ratio values only at the denominator sample points $\{\boldsymbol{x}_j^{\mathrm{de}}\}_{j=1}^{n_{\mathrm{de}}}$. Below, we consider the *induction* setup, where the entire density-ratio function $r^*(\boldsymbol{x})$ is estimated (Qin, 1998; Kanamori et al., 2012).

We use the following linear density-ratio model for density-ratio function learning:

$$r(\boldsymbol{x}) = \sum_{\ell=1}^{b} \theta_\ell \psi_\ell(\boldsymbol{x}) = \boldsymbol{\psi}(\boldsymbol{x})^\top \boldsymbol{\theta}, \tag{4}$$

where $\boldsymbol{\psi}(\boldsymbol{x}) : \mathbb{R}^d \to \mathbb{R}^b$ is a basis function vector and $\boldsymbol{\theta}$ $(\in \mathbb{R}^b)$ is a parameter vector. We assume that the basis functions are non-negative.

$$\boldsymbol{\psi}(\boldsymbol{x}) \geq \boldsymbol{0}_b.$$

Then model outputs at $\{\boldsymbol{x}_j^{\mathrm{de}}\}_{j=1}^{n_{\mathrm{de}}}$ are expressed in terms of the parameter vector $\boldsymbol{\theta}$ as

$$(r(\boldsymbol{x}_1^{\mathrm{de}}), \ldots, r(\boldsymbol{x}_{n_{\mathrm{de}}}^{\mathrm{de}}))^\top = \boldsymbol{\Psi}_{\mathrm{de}}^\top \boldsymbol{\theta},$$

where $\boldsymbol{\Psi}_{\mathrm{de}}$ is the $b \times n_{\mathrm{de}}$ *design matrix* defined by

$$\boldsymbol{\Psi}_{\mathrm{de}} := (\boldsymbol{\psi}(\boldsymbol{x}_1^{\mathrm{de}}), \ldots, \boldsymbol{\psi}(\boldsymbol{x}_{n_{\mathrm{de}}}^{\mathrm{de}})). \tag{5}$$

Then, following Eq.(2), the parameter $\boldsymbol{\theta}$ is learned as follows.

$$\widehat{\boldsymbol{\theta}} := \underset{\boldsymbol{\theta} \in \mathbb{R}^b}{\mathrm{argmin}} \left[\frac{1}{n_{\mathrm{de}}^2} \boldsymbol{\theta}^\top \boldsymbol{\Psi}_{\mathrm{de}} \boldsymbol{\Phi}_{\mathrm{de}}^\top \boldsymbol{\Phi}_{\mathrm{de}} \boldsymbol{\Psi}_{\mathrm{de}}^\top \boldsymbol{\theta} - \frac{2}{n_{\mathrm{de}} n_{\mathrm{nu}}} \boldsymbol{\theta}^\top \boldsymbol{\Psi}_{\mathrm{de}} \boldsymbol{\Phi}_{\mathrm{de}}^\top \boldsymbol{\Phi}_{\mathrm{nu}} \boldsymbol{1}_{n_{\mathrm{nu}}} \right]. \tag{6}$$

Taking the derivative of the above objective function with respect to $\boldsymbol{\theta}$ and setting it to zero, we have the solution $\widehat{\boldsymbol{\theta}}$ analytically as

$$\widehat{\boldsymbol{\theta}} = \frac{n_{\mathrm{de}}}{n_{\mathrm{nu}}} (\boldsymbol{\Psi}_{\mathrm{de}} \boldsymbol{\Phi}_{\mathrm{de}}^\top \boldsymbol{\Phi}_{\mathrm{de}} \boldsymbol{\Psi}_{\mathrm{de}}^\top)^{-1} \boldsymbol{\Psi}_{\mathrm{de}} \boldsymbol{\Phi}_{\mathrm{de}}^\top \boldsymbol{\Phi}_{\mathrm{nu}} \boldsymbol{1}_{n_{\mathrm{nu}}}.$$

One may include a normalization constraint, a non-negativity constraint (given that the basis functions are non-negative), and a regularization constraint to the optimization problem (6):

$$\frac{1}{n_{\mathrm{de}}}\mathbf{1}_{n_{\mathrm{de}}}^{\top}\boldsymbol{\Psi}_{\mathrm{de}}^{\top}\boldsymbol{\theta} = 1, \quad \boldsymbol{\theta} \geq \mathbf{0}_b, \text{ and } \boldsymbol{\theta} \leq B\mathbf{1}_b.$$

Then the optimization problem becomes a convex linearly-constrained quadratic program, whose solution can be obtained by a standard numerical solver.

The upper-bound parameter B, which works as a regularizer, may be optimized by *cross-validation* (CV) with respect to the moment-matching error MM defined by Eq.(1). Availability of CV would be one of the advantages of the inductive method (i.e., learning the entire density-ratio function).

2.1.2 Infinite-Order Approach: KMM

Matching a finite number of moments does not necessarily lead to the true density-ratio function $r^*(\boldsymbol{x})$, even if infinitely many samples are available. In order to guarantee that the true density-ratio function can always be obtained in the large-sample limit, all moments up to the infinite order need to be matched. Here we describe a method of infinite-oder moment-matching called *kernel mean matching* (KMM), which allows one to efficiently match all the moments using kernel functions (Huang et al., 2007; Gretton et al., 2009).

The basic idea of KMM is essentially the same as the finite-order approach, but a *universal reproducing kernel* $K(\boldsymbol{x}, \boldsymbol{x}')$ (Steinwart, 2001) is used as a non-linear transformation. The *Gaussian kernel*

$$K(\boldsymbol{x}, \boldsymbol{x}') = \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{x}'\|^2}{2\sigma^2}\right) \tag{7}$$

is an example of universal reproducing kernels. It has been shown that the solution of the following optimization problem agrees with the true density-ratio (Huang et al., 2007; Gretton et al., 2009):

$$\min_{r \in \mathcal{H}} \left\| \int K(\boldsymbol{x}, \cdot) p_{\mathrm{nu}}^*(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} - \int K(\boldsymbol{x}, \cdot) r(\boldsymbol{x}) p_{\mathrm{de}}^*(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \right\|_{\mathcal{H}}^2,$$

where $\mathcal{H}$ denotes a universal reproducing kernel Hilbert space and $\|\cdot\|_{\mathcal{H}}$ denotes its norm.

An empirical version of the above problem is expressed as

$$\min_{\boldsymbol{r} \in \mathbb{R}^{n_{\mathrm{de}}}} \left[\frac{1}{n_{\mathrm{de}}^2} \boldsymbol{r}^{\top} \boldsymbol{K}_{\mathrm{de,de}} \boldsymbol{r} - \frac{2}{n_{\mathrm{de}} n_{\mathrm{nu}}} \boldsymbol{r}^{\top} \boldsymbol{K}_{\mathrm{de,nu}} \mathbf{1}_{n_{\mathrm{nu}}} \right],$$

where $\boldsymbol{K}_{\mathrm{de,de}}$ and $\boldsymbol{K}_{\mathrm{de,nu}}$ denote the kernel Gram matrices defined by

$$[\boldsymbol{K}_{\mathrm{de,de}}]_{j,j'} = K(\boldsymbol{x}_j^{\mathrm{de}}, \boldsymbol{x}_{j'}^{\mathrm{de}}) \text{ and } [\boldsymbol{K}_{\mathrm{de,nu}}]_{j,i} = K(\boldsymbol{x}_j^{\mathrm{de}}, \boldsymbol{x}_i^{\mathrm{nu}}), \tag{8}$$

respectively. In the same way as the finite-order case, the solution can be obtained analytically as

$$\widehat{\boldsymbol{r}}_{\mathrm{de}} = \frac{n_{\mathrm{de}}}{n_{\mathrm{nu}}} \boldsymbol{K}_{\mathrm{de,de}}^{-1} \boldsymbol{K}_{\mathrm{de,nu}} \mathbf{1}_{n_{\mathrm{nu}}}. \tag{9}$$

If necessary, one may include a non-negativity constraint, a normalization constraint, and an upper bound in the same way as the finite-order case. Then the solution can be numerically obtained by solving a convex linearly-constrained quadratic programming problem.

For a linear density-ratio model (4), an inductive variant of KMM is formulated as

$$\min_{\boldsymbol{\theta} \in \mathbb{R}^b} \left[\frac{1}{n_{\mathrm{de}}^2} \boldsymbol{\theta}^\top \boldsymbol{\Psi}_{\mathrm{de}} \boldsymbol{K}_{\mathrm{de,de}} \boldsymbol{\Psi}_{\mathrm{de}}^\top \boldsymbol{\theta} - \frac{2}{n_{\mathrm{de}} n_{\mathrm{nu}}} \boldsymbol{\theta}^\top \boldsymbol{\Psi}_{\mathrm{de}} \boldsymbol{K}_{\mathrm{de,nu}} \mathbf{1}_{n_{\mathrm{nu}}} \right],$$

and the solution $\widehat{\boldsymbol{\theta}}$ is given by

$$\widehat{\boldsymbol{\theta}} = \frac{n_{\mathrm{de}}}{n_{\mathrm{nu}}} (\boldsymbol{\Psi}_{\mathrm{de}} \boldsymbol{K}_{\mathrm{de,de}} \boldsymbol{\Psi}_{\mathrm{de}})^{-1} \boldsymbol{\Psi}_{\mathrm{de}} \boldsymbol{K}_{\mathrm{de,nu}} \mathbf{1}_{n_{\mathrm{nu}}}.$$

2.1.3 Remarks

The infinite-order moment matching method, *kernel mean matching* (KMM), can efficiently match all the moments by making use of universal reproducing kernels. Indeed, KMM has an excellent theoretical property that it is consistent (Huang et al., 2007; Gretton et al., 2009). However, KMM has a limitation in model selection—there is no known method for determining the kernel parameter (i.e., the Gaussian kernel width). A popular heuristic of setting the Gaussian width to the median distance between samples (Schölkopf and Smola, 2002) would be useful in some cases, but this may not always be reasonable.

In the above, moment matching was performed in terms of the squared norm, which led to an analytic-form solution (if no constraint is imposed). As shown in Kanamori et al. (2012), moment matching can be systematically generalized to various divergences.

2.2 Probabilistic Classification

Here, we describe a framework of density-ratio estimation through *probabilistic classification.*

2.2.1 Basic Framework

The basic idea of the probabilistic classification approach is to obtain a probabilistic classifier that separates numerator samples $\{\boldsymbol{x}_i^{\mathrm{nu}}\}_{i=1}^{n_{\mathrm{nu}}}$ and denominator samples $\{\boldsymbol{x}_j^{\mathrm{de}}\}_{j=1}^{n_{\mathrm{de}}}$.

Let us assign a label $y = +1$ to $\{\boldsymbol{x}_i^{\mathrm{nu}}\}_{i=1}^{n_{\mathrm{nu}}}$ and $y = -1$ to $\{\boldsymbol{x}_j^{\mathrm{de}}\}_{j=1}^{n_{\mathrm{de}}}$, respectively. Then the two densities $p_{\mathrm{nu}}^*(\boldsymbol{x})$ and $p_{\mathrm{de}}^*(\boldsymbol{x})$ are written as

$$p_{\mathrm{nu}}^*(\boldsymbol{x}) = p^*(\boldsymbol{x}|y = +1) \quad \text{and} \quad p_{\mathrm{de}}^*(\boldsymbol{x}) = p^*(\boldsymbol{x}|y = -1),$$

respectively. Note that y is regarded as a random variable here. An application of Bayes' theorem,

$$p^*(\boldsymbol{x}|y) = \frac{p^*(y|\boldsymbol{x})p^*(\boldsymbol{x})}{p^*(y)},$$

yields that the density-ratio $r^*(\boldsymbol{x})$ can be expressed in terms of y as follows:

$$\begin{aligned} r^*(\boldsymbol{x}) = \frac{p^*_{\mathrm{nu}}(\boldsymbol{x})}{p^*_{\mathrm{de}}(\boldsymbol{x})} &= \left(\frac{p^*(y=+1|\boldsymbol{x})p^*(\boldsymbol{x})}{p^*(y=+1)}\right)\left(\frac{p^*(y=-1|\boldsymbol{x})p^*(\boldsymbol{x})}{p^*(y=-1)}\right)^{-1} \\ &= \frac{p^*(y=-1)}{p^*(y=+1)}\frac{p^*(y=+1|\boldsymbol{x})}{p^*(y=-1|\boldsymbol{x})}. \end{aligned}$$

The ratio $p^*(y=-1)/p^*(y=+1)$ may be simply approximated by the ratio of the sample size:

$$\frac{p^*(y=-1)}{p^*(y=+1)} \approx \frac{n_{\mathrm{de}}/(n_{\mathrm{de}}+n_{\mathrm{nu}})}{n_{\mathrm{nu}}/(n_{\mathrm{de}}+n_{\mathrm{nu}})} = \frac{n_{\mathrm{de}}}{n_{\mathrm{nu}}}.$$

The 'class'-posterior probability $p^*(y|\boldsymbol{x})$ may be approximated by separating $\{\boldsymbol{x}_i^{\mathrm{nu}}\}_{i=1}^{n_{\mathrm{nu}}}$ and $\{\boldsymbol{x}_j^{\mathrm{de}}\}_{j=1}^{n_{\mathrm{de}}}$ using a probabilistic classifier. Thus, given an estimator of the class-posterior probability, $\widehat{p}(y|\boldsymbol{x})$, a density-ratio estimator $\widehat{r}(\boldsymbol{x})$ can be constructed as

$$\widehat{r}(\boldsymbol{x}) = \frac{n_{\mathrm{de}}}{n_{\mathrm{nu}}}\frac{\widehat{p}(y=+1|\boldsymbol{x})}{\widehat{p}(y=-1|\boldsymbol{x})}. \tag{10}$$

A practical advantage of the probabilistic classification approach would be its easy implementability. Indeed, one can directly use standard probabilistic classification algorithms for density-ratio estimation. Another, more important advantage of the probabilistic classification approach is that model selection (i.e., tuning the basis functions and the regularization parameter) is possible by standard *cross-validation* since the estimation problem involved in this framework is a standard supervised classification problem.

Below, two probabilistic classification algorithms are described. For making the explanation simple, we consider a set of paired samples $\{(\boldsymbol{x}_k, y_k)\}_{k=1}^n$, where, for $n = n_{\mathrm{nu}} + n_{\mathrm{de}}$,

$$\begin{aligned} (\boldsymbol{x}_1,\ldots,\boldsymbol{x}_n) &:= (\boldsymbol{x}_1^{\mathrm{nu}},\ldots,\boldsymbol{x}_{n_{\mathrm{nu}}}^{\mathrm{nu}},\boldsymbol{x}_1^{\mathrm{de}},\ldots,\boldsymbol{x}_{n_{\mathrm{de}}}^{\mathrm{de}}), \\ (y_1,\ldots,y_n) &:= (\underbrace{+1,\ldots,+1}_{n_{\mathrm{nu}}},\underbrace{-1,\ldots,-1}_{n_{\mathrm{de}}}). \end{aligned}$$

2.2.2 Logistic Regression

Here, a popular probabilistic classification algorithm called *logistic regression* (Hastie et al., 2001) is explained.

A logistic regression classifier employs a parametric model of the following form for expressing the class-posterior probability $p^*(y|\boldsymbol{x})$,

$$p(y|\boldsymbol{x};\boldsymbol{\theta}) = \frac{1}{1+\exp\left(-y\boldsymbol{\psi}(\boldsymbol{x})^\top\boldsymbol{\theta}\right)},$$

where $\boldsymbol{\psi}(\boldsymbol{x}) : \mathbb{R}^d \rightarrow \mathbb{R}^b$ is a basis function vector and $\boldsymbol{\theta}$ $(\in \mathbb{R}^b)$ is a parameter vector. The parameter vector $\boldsymbol{\theta}$ is determined so that the *penalized log-likelihood* is maximized, which can be expressed as the following minimization problem:

$$\widehat{\boldsymbol{\theta}} := \mathop{\mathrm{argmin}}_{\boldsymbol{\theta}\in\mathbb{R}^b} \left[\sum_{k=1}^{n} \log\left(1 + \exp\left(-y_k \boldsymbol{\psi}(\boldsymbol{x}_k)^\top \boldsymbol{\theta}\right)\right) + \lambda \boldsymbol{\theta}^\top \boldsymbol{\theta} \right], \tag{11}$$

where $\lambda \boldsymbol{\theta}^\top \boldsymbol{\theta}$ is a penalty term included for regularization purposes.

Since the objective function in Eq.(11) is convex, the global optimal solution can be obtained by a standard non-linear optimization technique such as the *gradient descent method* or *(quasi-)Newton methods* (Hastie et al., 2001; Minka, 2007). Finally, a density-ratio estimator $\widehat{r}_{\mathrm{LR}}(\boldsymbol{x})$ is given by

$$\widehat{r}_{\mathrm{LR}}(\boldsymbol{x}) = \frac{n_{\mathrm{de}}}{n_{\mathrm{nu}}} \frac{1 + \exp\left(\boldsymbol{\psi}(\boldsymbol{x})^\top \widehat{\boldsymbol{\theta}}\right)}{1 + \exp\left(-\boldsymbol{\psi}(\boldsymbol{x})^\top \widehat{\boldsymbol{\theta}}\right)} = \frac{n_{\mathrm{de}}}{n_{\mathrm{nu}}} \exp\left(\boldsymbol{\psi}(\boldsymbol{x})^\top \widehat{\boldsymbol{\theta}}\right),$$

where ‘LR’ stands for ‘logistic regression’.

Suppose that the logistic regression model $p(y|\boldsymbol{x};\boldsymbol{\theta})$ satisfies the following two conditions:

- The constant function is included in the basis functions, i.e., there exists $\boldsymbol{\theta}^\circ$ such that

$$\boldsymbol{\psi}(\boldsymbol{x})^\top \boldsymbol{\theta}^\circ = 1.$$

- The model is *correctly specified*, i.e., there exists $\boldsymbol{\theta}^*$ such that

$$p(y|\boldsymbol{x};\boldsymbol{\theta}^*) = p^*(y|\boldsymbol{x}).$$

Then it was proved that the logistic regression approach is optimal among a class of semi-parametric estimators in the sense that the asymptotic variance is minimized (Qin, 1998). However, when the model is misspecified (which would be the case in practice), the density matching approach explained in Section 2.3 would be more preferable (Kanamori et al., 2010).

When *multi-class* logistic regression classifiers are used, density-ratios among multiple densities can be estimated simultaneously (Bickel et al., 2008). This is useful, e.g., for solving *multi-task learning* problems (Caruana et al., 1997).

2.2.3 Least-Squares Probabilistic Classifier

Although the performance of these general-purpose non-linear optimization techniques has been improved together with the evolution of computer environment in the last decade, training logistic regression classifiers is still computationally expensive. Here, an alternative probabilistic classification algorithm called *least-squares probabilistic classifier* (LSPC; Sugiyama, 2010) is described. LSPC is computationally more efficient than logistic regression, with comparable accuracy in practice.

In LSPC, the class-posterior probability $p^*(y|\boldsymbol{x})$ is modeled as

$$p(y|\boldsymbol{x};\boldsymbol{\theta}) := \sum_{\ell=1}^{b} \theta_\ell \psi(\boldsymbol{x}, y) \ = \boldsymbol{\psi}(\boldsymbol{x}, y)^\top \boldsymbol{\theta},$$

where $\boldsymbol{\psi}(\boldsymbol{x}, y)$ $(\in \mathbb{R}^b)$ is a non-negative basis function vector, and $\boldsymbol{\theta}$ $(\in \mathbb{R}^b)$ is a parameter vector. The class label y takes a value in $\{1, \ldots, c\}$, where c is the number of classes.

The basic idea of LSPC is to express the class-posterior probability $p^*(y|\boldsymbol{x})$ in terms of the equivalent density-ratio expression: $p^*(\boldsymbol{x}, y)/p^*(\boldsymbol{x})$. Then the density-ratio estimation method called *unconstrained least-squares importance fitting* (uLSIF; Kanamori et al., 2009) is used for estimating this density-ratio. Since uLSIF will be reviewed in detail in Section 2.4.3, we only describe the final solution here.

Let

$$\widehat{\boldsymbol{H}} := \frac{1}{n}\sum_{k=1}^{n}\sum_{y=1}^{c} \boldsymbol{\psi}(\boldsymbol{x}_k, y)\boldsymbol{\psi}(\boldsymbol{x}_k, y)^\top \quad \text{and} \quad \widehat{\boldsymbol{h}} := \frac{1}{n}\sum_{k=1}^{n} \boldsymbol{\psi}(\boldsymbol{x}_k, y_k).$$

Then the uLSIF solution is given analytically as $\widehat{\boldsymbol{\theta}} = (\widehat{\boldsymbol{H}} + \lambda \boldsymbol{I}_b)^{-1}\widehat{\boldsymbol{h}}$, where λ (≥ 0) is the regularization parameter and $\boldsymbol{I}_b$ is the b-dimensional identity matrix. In order to assure that the output of LSPC is a probability, the outputs are normalized and negative outputs are rounded up to zero (Yamada et al., 2011):

$$\widehat{p}(y|\boldsymbol{x}) = \frac{\max(0, \boldsymbol{\psi}(\boldsymbol{x}, y)^\top \widehat{\boldsymbol{\theta}})}{\sum_{y'=1}^{c} \max(0, \boldsymbol{\psi}(\boldsymbol{x}, y')^\top \widehat{\boldsymbol{\theta}})}.$$

A standard choice of basis functions $\boldsymbol{\psi}(\boldsymbol{x}, y)$ would be a *kernel* model:

$$p(y|\boldsymbol{x};\boldsymbol{\theta}) = \sum_{\ell=1}^{n} \theta_\ell^{(y)} K(\boldsymbol{x}, \boldsymbol{x}_\ell), \tag{12}$$

where $K(\boldsymbol{x}, \boldsymbol{x}')$ is some kernel function such as the *Gaussian kernel* (7). Then the matrix $\widehat{\boldsymbol{H}}$ becomes block-diagonal. Thus, we only need to train a model with n parameters separately c times for each class $y = 1, \ldots, c$. Since all the diagonal block matrices are the same, the computational complexity for computing the solution is $\mathcal{O}(n^3 + cn^2)$.

Let us further reduce the number of kernels in model (12). To this end, we focus on a kernel function $K(\boldsymbol{x}, \boldsymbol{x}')$ that is "localized". Examples of such localized kernels include the popular Gaussian kernel. The idea is to reduce the number of kernels by locating the kernels only at samples belonging to the *target* class:

$$p(y|\boldsymbol{x};\boldsymbol{\theta}) = \sum_{\ell=1}^{n_y} \theta_\ell^{(y)} K(\boldsymbol{x}, \boldsymbol{x}_\ell^{(y)}), \tag{13}$$

where n_y is the number of training samples in class y and $\{\boldsymbol{x}_k^{(y)}\}_{k=1}^{n_y}$ is the training input samples in class y. The rationale behind this model simplification is as follows. By definition, the class-posterior probability $p^*(y|\boldsymbol{x})$ takes large values in the regions where samples in class y are dense; conversely, $p^*(y|\boldsymbol{x})$ takes smaller values (i.e., close to zero) in the regions where samples in class y are sparse. When a non-negative function is approximated by a localized kernel model, many kernels may be needed in the region where the output of the target function is large; on the other hand, only a small number of kernels would be enough in the region where the output of the target function is close to zero. Following this heuristic, many kernels are allocated in the region where $p^*(y|\boldsymbol{x})$ takes large values, which can be achieved by Eq.(13).

This model simplification allows one to further reduce the computational cost since the size of the target blocks in matrix $\widehat{\boldsymbol{H}}$ is further reduced. In order to determine the n_y-dimensional parameter vector $\boldsymbol{\theta}^{(y)} = (\theta_1^{(y)}, \ldots, \theta_{n_y}^{(y)})^\top$ for each class y, we only need to solve the following system of n_y linear equations:

$$(\widehat{\boldsymbol{H}}^{(y)} + \lambda \boldsymbol{I}_{n_y})\boldsymbol{\theta}^{(y)} = \widehat{\boldsymbol{h}}^{(y)}, \tag{14}$$

where $\widehat{\boldsymbol{H}}^{(y)}$ is the $n_y \times n_y$ matrix, and $\widehat{\boldsymbol{h}}^{(y)}$ is the n_y-dimensional vector defined as

$$\widehat{H}_{\ell,\ell'}^{(y)} := \frac{1}{n_y}\sum_{k=1}^{n_y} K(\boldsymbol{x}_k^{(y)}, \boldsymbol{x}_\ell^{(y)})K(\boldsymbol{x}_k^{(y)}, \boldsymbol{x}_{\ell'}^{(y)}) \quad \text{and} \quad \widehat{h}_\ell^{(y)} := \frac{1}{n_y}\sum_{k=1}^{n_y} K(\boldsymbol{x}_k^{(y)}, \boldsymbol{x}_\ell^{(y)}).$$

Let $\widehat{\boldsymbol{\theta}}^{(y)}$ be the solution of Eq.(14). Then the final solution is given by

$$\widehat{p}(y|\boldsymbol{x}) = \frac{\max\left(0, \sum_{\ell=1}^{n_y} \widehat{\theta}_\ell^{(y)} K(\boldsymbol{x}, \boldsymbol{x}_\ell^{(y)})\right)}{\sum_{y'=1}^{c} \max\left(0, \sum_{\ell=1}^{n_{y'}} \widehat{\theta}_\ell^{(y')} K(\boldsymbol{x}, \boldsymbol{x}_\ell^{(y')})\right)}. \tag{15}$$

For the simplified model (13), the computational complexity for computing the solution is $\mathcal{O}(cn_y^3)$—when $n_y = n/c$ for all y, this is equal to $\mathcal{O}(c^{-2}n^3)$. Thus, this approach is computationally highly efficient for multi-class problems with large c.

A MATLAB® implementation of LSPC is available from

http://sugiyama-www.cs.titech.ac.jp/~sugi/software/LSPC/

2.2.4 Remarks

Density-ratio estimation by probabilistic classification can successfully avoid density estimation by casting the problem of density-ratio estimation as the problem of estimating the 'class'-posterior probability. An advantage of the probabilistic classification approach over the moment matching approach explained in Section 2.1 is that cross-validation can be used for model selection. Furthermore, existing software packages of probabilistic classification algorithms can be directly used for density-ratio estimation.

The probabilistic classification approach with logistic regression was shown to have a suitable theoretical property (Qin, 1998): if the logistic regression model is *correctly specified*, the probabilistic classification approach is optimal among a broad class of semi-parametric estimators. However, this strong theoretical property is not true when the correct model assumption is not fulfilled.

An advantage of the probabilistic classification approach is that it can be used for estimating density-ratios among multiple densities by multi-class probabilistic classifiers. In this context, the *least-squares probabilistic classifier* (LSPC) would be practically useful due to its computational efficiency.

2.3 Density Matching

Here, we describe a framework of density-ratio estimation by *density matching* under the KL divergence.

2.3.1 Basic Framework

Let $r(\boldsymbol{x})$ be a model of the true density-ratio $r^*(\boldsymbol{x}) = p^*_{\mathrm{nu}}(\boldsymbol{x})/p^*_{\mathrm{de}}(\boldsymbol{x})$. Then the numerator density $p^*_{\mathrm{nu}}(\boldsymbol{x})$ may be modeled by $p_{\mathrm{nu}}(\boldsymbol{x}) = r(\boldsymbol{x})p^*_{\mathrm{de}}(\boldsymbol{x})$. Now let us consider the KL divergence from $p^*_{\mathrm{nu}}(\boldsymbol{x})$ to $p_{\mathrm{nu}}(\boldsymbol{x})$:

$$\mathrm{KL}'(p^*_{\mathrm{nu}} \| p_{\mathrm{nu}}) := \int p^*_{\mathrm{nu}}(\boldsymbol{x}) \log \frac{p^*_{\mathrm{nu}}(\boldsymbol{x})}{p_{\mathrm{nu}}(\boldsymbol{x})} \mathrm{d}\boldsymbol{x} = C - \mathrm{KL}(r),$$

where $C := \int p^*_{\mathrm{nu}}(\boldsymbol{x}) \log \frac{p^*_{\mathrm{nu}}(\boldsymbol{x})}{p^*_{\mathrm{de}}(\boldsymbol{x})} \mathrm{d}\boldsymbol{x}$ is a constant irrelevant to r, and $\mathrm{KL}(r)$ is the relevant part:

$$\mathrm{KL}(r) := \int p^*_{\mathrm{nu}}(\boldsymbol{x}) \log r(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \approx \frac{1}{n_{\mathrm{nu}}} \sum_{i=1}^{n_{\mathrm{nu}}} \log r(\boldsymbol{x}^{\mathrm{nu}}_i).$$

Since $p_{\mathrm{nu}}(\boldsymbol{x})$ is a probability density function, its integral should be one:

$$1 = \int p_{\mathrm{nu}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} = \int r(\boldsymbol{x}) p^*_{\mathrm{de}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \approx \frac{1}{n_{\mathrm{de}}} \sum_{j=1}^{n_{\mathrm{de}}} r(\boldsymbol{x}^{\mathrm{de}}_j).$$

Furthermore, the density $p_{\mathrm{nu}}(\boldsymbol{x})$ should be non-negative, which can be achieved by $r(\boldsymbol{x}) \geq 0$ for all $\boldsymbol{x}$. Combining these equations together, we have the following optimization problem.

$$\begin{aligned} &\max_{r} \ \frac{1}{n_{\mathrm{nu}}} \sum_{i=1}^{n_{\mathrm{nu}}} \log r(\boldsymbol{x}^{\mathrm{nu}}_i) \\ &\text{s.t.} \ \frac{1}{n_{\mathrm{de}}} \sum_{j=1}^{n_{\mathrm{de}}} r(\boldsymbol{x}^{\mathrm{de}}_j) = 1 \text{ and } r(\boldsymbol{x}) \geq 0 \text{ for all } \boldsymbol{x}. \end{aligned}$$

This formulation is called the *KL importance estimation procedure* (KLIEP; Sugiyama et al., 2008).

Possible hyper-parameters in KLIEP (such as basis parameters and regularization parameters) can be optimized using *cross-validation* with respect to the KL divergence, where the numerator samples $\{\boldsymbol{x}_i^{\mathrm{nu}}\}_{i=1}^{n_{\mathrm{nu}}}$ appearing in the objective function may only be cross-validated (Sugiyama et al., 2008).

Below, practical implementations of KLIEP for various density-ratio models are described.

2.3.2 Linear and Kernel Models

Let us employ a linear model for density-ratio estimation.

$$r(\boldsymbol{x}) = \sum_{\ell=1}^{b} \theta_\ell \psi_\ell(\boldsymbol{x}) = \boldsymbol{\psi}(\boldsymbol{x})^\top \boldsymbol{\theta}, \tag{16}$$

where $\boldsymbol{\psi}(\boldsymbol{x}) : \mathbb{R}^d \to \mathbb{R}^b$ is a non-negative basis function vector, and $\boldsymbol{\theta}$ $(\in \mathbb{R}^b)$ is a parameter vector. Then the KLIEP optimization problem for the linear model is expressed as follows (Sugiyama et al., 2008).

$$\max_{\boldsymbol{\theta}\in\mathbb{R}^b} \frac{1}{n_{\mathrm{nu}}} \sum_{i=1}^{n_{\mathrm{nu}}} \log(\boldsymbol{\psi}(\boldsymbol{x}_i^{\mathrm{nu}})^\top \boldsymbol{\theta}) \quad \text{s.t.} \quad \overline{\boldsymbol{\psi}}_{\mathrm{de}}^\top \boldsymbol{\theta} = 1 \text{ and } \boldsymbol{\theta} \geq \mathbf{0}_b,$$

where $\overline{\boldsymbol{\psi}}_{\mathrm{de}} := \frac{1}{n_{\mathrm{de}}} \sum_{j=1}^{n_{\mathrm{de}}} \boldsymbol{\psi}(\boldsymbol{x}_j^{\mathrm{de}})$.

Since the above optimization problem is *convex*, there exists the unique global optimum solution. Furthermore, the KLIEP solution tends to be *sparse*, i.e., many parameters take exactly zero, because of the non-negativity constraint. Such sparsity would contribute to reducing the computation time when computing estimated density-ratio values. As can be confirmed from the above optimization problem, the denominator samples $\{\boldsymbol{x}_j^{\mathrm{de}}\}_{j=1}^{n_{\mathrm{de}}}$ appear only in terms of the basis-transformed mean $\overline{\boldsymbol{\psi}}_{\mathrm{de}}$. Thus, KLIEP for linear models is computationally efficient even when the number n_{de} of denominator samples is very large.

The performance of KLIEP depends on the choice of the basis functions $\boldsymbol{\psi}(\boldsymbol{x})$. As explained below, the use of the following Gaussian kernel model would be reasonable:

$$r(\boldsymbol{x}) = \sum_{\ell=1}^{n_{\mathrm{nu}}} \theta_\ell K(\boldsymbol{x}, \boldsymbol{x}_\ell^{\mathrm{nu}}), \tag{17}$$

where $K(\boldsymbol{x}, \boldsymbol{x}')$ is the Gaussian kernel (7). The reason why the numerator samples $\{\boldsymbol{x}_i^{\mathrm{nu}}\}_{i=1}^{n_{\mathrm{nu}}}$, not the denominator samples $\{\boldsymbol{x}_j^{\mathrm{de}}\}_{j=1}^{n_{\mathrm{de}}}$, are chosen as the Gaussian centers is as follows. By definition, the density-ratio $r^*(\boldsymbol{x})$ tends to take large values if $p_{\mathrm{de}}^*(\boldsymbol{x})$ is small and $p_{\mathrm{nu}}^*(\boldsymbol{x})$ is large. Conversely, $r^*(\boldsymbol{x})$ tends to be small (i.e., close to zero) if $p_{\mathrm{de}}^*(\boldsymbol{x})$ is large and $p_{\mathrm{nu}}^*(\boldsymbol{x})$ is small. When a non-negative function is approximated by a

Gaussian kernel model, many kernels may be needed in the region where the output of the target function is large. On the other hand, only a small number of kernels would be enough in the region where the output of the target function is close to zero. Following this heuristic, many kernels are allocated in the region where $p^*_{\mathrm{nu}}(\boldsymbol{x})$ takes large values, which can be achieved by setting the Gaussian centers at $\{\boldsymbol{x}_i^{\mathrm{nu}}\}_{i=1}^{n_{\mathrm{nu}}}$.

The KLIEP methods for linear/kernel models are referred to as *linear KLIEP* (L-KLIEP) and *kernel KLIEP* (K-KLIEP), respectively. A MATLAB® implementation of the K-KLIEP algorithm is available from

http://sugiyama-www.cs.titech.ac.jp/˜sugi/software/KLIEP/

2.3.3 Log-Linear Models

Another popular model choice would be the *log-linear model* (Tsuboi et al., 2009; Kanamori et al., 2010):

$$r(\boldsymbol{x};\boldsymbol{\theta},\theta_0)=\exp\left(\boldsymbol{\psi}(\boldsymbol{x})^\top\boldsymbol{\theta}+\theta_0\right), \tag{18}$$

where θ_0 is a normalization parameter. From the normalization constraint

$$\frac{1}{n_{\mathrm{de}}}\sum_{j=1}^{n_{\mathrm{de}}} r(\boldsymbol{x}_j^{\mathrm{de}};\boldsymbol{\theta},\theta_0)=1,$$

θ_0 is determined as

$$\widehat{\theta}_0=-\log\left(\frac{1}{n_{\mathrm{de}}}\sum_{j=1}^{n_{\mathrm{de}}}\exp\left(\boldsymbol{\psi}(\boldsymbol{x}_j^{\mathrm{de}})^\top\boldsymbol{\theta}\right)\right).$$

Then the density-ratio model is expressed as

$$r(\boldsymbol{x};\boldsymbol{\theta})=\frac{\exp\left(\boldsymbol{\psi}(\boldsymbol{x})^\top\boldsymbol{\theta}\right)}{\frac{1}{n_{\mathrm{de}}}\sum_{j=1}^{n_{\mathrm{de}}}\exp\left(\boldsymbol{\psi}(\boldsymbol{x}_j^{\mathrm{de}})^\top\boldsymbol{\theta}\right)}.$$

By definition, outputs of the log-linear model $r(\boldsymbol{x};\boldsymbol{\theta})$ are non-negative for all $\boldsymbol{x}$. Thus, we do not need the non-negativity constraint on the parameter. Then the KLIEP optimization criterion is expressed as

$$\max_{\boldsymbol{\theta}\in\mathbb{R}^b}\left[\overline{\boldsymbol{\psi}}_{\mathrm{nu}}^\top\boldsymbol{\theta}-\log\left(\frac{1}{n_{\mathrm{de}}}\sum_{j=1}^{n_{\mathrm{de}}}\exp(\boldsymbol{\psi}(\boldsymbol{x}_j^{\mathrm{de}})^\top\boldsymbol{\theta})\right)\right],$$

where $\overline{\boldsymbol{\psi}}_{\mathrm{nu}}:=\frac{1}{n_{\mathrm{nu}}}\sum_{i=1}^{n_{\mathrm{nu}}}\boldsymbol{\psi}(\boldsymbol{x}_i^{\mathrm{nu}})$. This is an unconstrained convex optimization problem, so the global optimal solution can be obtained by, e.g., the gradient method and (quasi-)Newton methods. Since the numerator samples $\{\boldsymbol{x}_i^{\mathrm{nu}}\}_{i=1}^{n_{\mathrm{nu}}}$ appear only in terms of the basis-transformed mean $\overline{\boldsymbol{\psi}}_{\mathrm{nu}}$, KLIEP for log-linear models is computationally efficient even when the number n_{nu} of numerator samples is very large (cf. KLIEP for linear/kernel models is computationally efficient when n_{de} is very large; see Section 2.3.2).

The KLIEP method for log-linear models is called *log-linear KLIEP* (LL-KLIEP).

2.3.4 Gaussian Mixture Models

In the Gaussian kernel model (17), the Gaussian shape is spherical and its width is controlled by a single width parameter σ. It is possible to use correlated Gaussian kernels, but choosing the covariance matrix via cross-validation would be computationally intractable.

Another option is to also estimate the covariance matrix directly from data. For this purpose, the *Gaussian mixture model* comes in handy (Yamada and Sugiyama, 2009):

$$r(\boldsymbol{x}; \{\theta_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}_{k=1}^{c}) = \sum_{k=1}^{c} \theta_k K(\boldsymbol{x}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k), \tag{19}$$

where c is the number of mixing components, $\{\theta_k\}_{k=1}^{c}$ are mixing coefficients, $\{\boldsymbol{\mu}_k\}_{k=1}^{c}$ are means of Gaussian functions, $\{\boldsymbol{\Sigma}_k\}_{k=1}^{c}$ are covariance matrices of Gaussian functions, and $K(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma})$ is the Gaussian kernel with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$:

$$K(\boldsymbol{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) := \exp\left(-\frac{1}{2}(\boldsymbol{x} - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1} (\boldsymbol{x} - \boldsymbol{\mu})\right). \tag{20}$$

Note that $\boldsymbol{\Sigma}$ should be *positive definite*, i.e., all the eigenvectors of $\boldsymbol{\Sigma}$ should be strictly positive.

For the Gaussian mixture model (19), the KLIEP optimization problem is expressed as

$$\begin{aligned}
\max_{\{\theta_k, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k\}_{k=1}^{c}} \quad & \frac{1}{n_{\mathrm{nu}}} \sum_{i=1}^{n_{\mathrm{nu}}} \log\left(\sum_{k=1}^{c} \theta_k K(\boldsymbol{x}_i^{\mathrm{nu}}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)\right) \\
\text{s.t.} \quad & \frac{1}{n_{\mathrm{de}}} \sum_{j=1}^{n_{\mathrm{de}}} \sum_{k=1}^{c} \theta_k K(\boldsymbol{x}_j^{\mathrm{de}}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) = 1, \\
& \theta_k \geq 0 \text{ and } \boldsymbol{\Sigma}_k \succ \boldsymbol{O} \text{ for } k = 1, \ldots, c,
\end{aligned}$$

where $\boldsymbol{\Sigma}_k \succ \boldsymbol{O}$ means that $\boldsymbol{\Sigma}_k$ is positive definite.

The above optimization problem is *non-convex*, and there is no known method for obtaining the global optimal solution. In practice, a local optimal solution may be numerically obtained by, e.g., a fixed-point method.

The KLIEP method for Gaussian mixture models is called *Gaussian-mixture KLIEP* (GM-KLIEP).

2.3.5 Probabilistic PCA Mixture Models

The Gaussian mixture model explained above would be more flexible than linear/kernel/log-linear models and suitable for approximating correlated density-ratio functions. However, when the target density-ratio function is (locally) rank-deficient, its behavior could be unstable since inverse covariance matrices are included in the Gaussian function (see Eq.(20)). To cope with this problem, the use of *a mixture of probabilistic principal component analyzers* (PPCA; Tipping and Bishop, 1999) was proposed for density-ratio estimation (Yamada et al., 2010).

The PPCA mixture model is defined as

$$r(\boldsymbol{x}; \{\theta_k, \boldsymbol{\mu}_k, \sigma_k^2, \boldsymbol{W}_k\}_{k=1}^c) = \sum_{k=1}^c \theta_k K(\boldsymbol{x}; \boldsymbol{\mu}_k, \sigma_k^2, \boldsymbol{W}_k),$$

where c is the number of mixing components and $\{\theta_k\}_{k=1}^c$ are mixing coefficients. $K(\boldsymbol{x}; \boldsymbol{\mu}, \sigma^2, \boldsymbol{W})$ is a PPCA model defined by

$$K(\boldsymbol{x}; \boldsymbol{\mu}, \sigma^2, \boldsymbol{W}) = (2\pi\sigma^2)^{-\frac{d}{2}} \det(\boldsymbol{C})^{-\frac{1}{2}} \exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^\top \boldsymbol{C}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right),$$

where 'det' denotes the determinant, $\boldsymbol{\mu}$ is the mean of the Gaussian function, σ^2 is the variance of the Gaussian function, $\boldsymbol{W}$ is a $d \times m$ 'projection' matrix onto a m-dimensional *latent* space (where $m \leq d$), and $\boldsymbol{C} = \boldsymbol{W}\boldsymbol{W}^\top + \sigma^2 \boldsymbol{I}_d$.

Then the KLIEP optimization criterion is expressed as

$$\begin{aligned}
\max_{\{\theta_k, \boldsymbol{\mu}_k, \sigma_k^2, \boldsymbol{W}_k\}_{k=1}^c} \quad & \frac{1}{n_{\mathrm{nu}}} \sum_{i=1}^{n_{\mathrm{nu}}} \log\left(\sum_{k=1}^c \theta_k K(\boldsymbol{x}_i^{\mathrm{nu}}; \boldsymbol{\mu}_k, \sigma_k^2, \boldsymbol{W}_k)\right) \\
\text{s.t.} \quad & \frac{1}{n_{\mathrm{de}}} \sum_{j=1}^{n_{\mathrm{de}}} \sum_{k=1}^c \theta_k K(\boldsymbol{x}_j^{\mathrm{de}}; \boldsymbol{\mu}_k, \sigma_k^2, \boldsymbol{W}_k) = 1, \\
& \theta_k \geq 0 \text{ for } k = 1, \ldots, c.
\end{aligned}$$

The above optimization is non-convex, so a local optimal solution may be found by some algorithm in practice. When the dimensionality of the latent space, m, is equal to the entire dimensionality d, PPCA models are reduced to ordinary Gaussian models. Thus, PPCA models can be regarded as an extension of Gaussian models to (locally) rank-deficient data.

The KLIEP method for PPCA mixture models is called *PPCA-mixture KLIEP* (PM-KLIEP).

2.3.6 Remarks

Density-ratio estimation by density matching under the KL divergence allows one to avoid density estimation when estimating density-ratios (Section 2.3.1). Furthermore, cross-validation with respect to the KL divergence is available for model selection.

The method, called the *KL importance estimation procedure* (KLIEP), is applicable to a variety of models such as linear models, kernel models, log-linear models, Gaussian mixture models, and probabilistic principal-component-analyzer mixture models.

2.4 Density-Ratio Fitting

Here, we describe a framework of density-ratio estimation by *least-squares density-ratio fitting* (Kanamori et al., 2009).

2.4.1 Basic Framework

The model $r(\boldsymbol{x})$ of the true density-ratio function $r^*(\boldsymbol{x}) = p^*_{\mathrm{nu}}(\boldsymbol{x})/p^*_{\mathrm{de}}(\boldsymbol{x})$ is learned so that the following squared error SQ' is minimized:

$$\begin{aligned}\mathrm{SQ}'(r) &:= \frac{1}{2}\int (r(\boldsymbol{x}) - r^*(\boldsymbol{x}))^2 \, p^*_{\mathrm{de}}(\boldsymbol{x})\mathrm{d}\boldsymbol{x}.\\ &= \frac{1}{2}\int r(\boldsymbol{x})^2 p^*_{\mathrm{de}}(\boldsymbol{x})\mathrm{d}\boldsymbol{x} - \int r(\boldsymbol{x}) p^*_{\mathrm{nu}}(\boldsymbol{x})\mathrm{d}\boldsymbol{x} + \frac{1}{2}\int r^*(\boldsymbol{x}) p^*_{\mathrm{nu}}(\boldsymbol{x})\mathrm{d}\boldsymbol{x},\end{aligned}$$

where the last term is a constant and therefore can be safely ignored. Let us denote the first two terms by SQ:

$$\mathrm{SQ}(r) := \frac{1}{2}\int r(\boldsymbol{x})^2 p^*_{\mathrm{de}}(\boldsymbol{x})\mathrm{d}\boldsymbol{x} - \int r(\boldsymbol{x}) p^*_{\mathrm{nu}}(\boldsymbol{x})\mathrm{d}\boldsymbol{x}.$$

Approximating the expectations in SQ by empirical averages, we obtain the following optimization problem:

$$\min_{r} \left[\sum_{j=1}^{n_{\mathrm{de}}} r(\boldsymbol{x}_j^{\mathrm{de}})^2 - \frac{1}{n_{\mathrm{nu}}} \sum_{i=1}^{n_{\mathrm{nu}}} r(\boldsymbol{x}_i^{\mathrm{nu}}) \right]. \tag{21}$$

We refer to this formulation as *least-squares importance fitting* (LSIF). Possible hyper-parameters (such as basis parameters and regularization parameters) can be optimized by *cross-validation* with respect to the SQ criterion (Kanamori et al., 2009).

Below, two implementations of LSIF for the following linear/kernel models are described:

$$r(\boldsymbol{x}) = \sum_{\ell=1}^{b} \theta_\ell \psi_\ell(\boldsymbol{x}) = \boldsymbol{\psi}(\boldsymbol{x})^\top \boldsymbol{\theta},$$

where $\boldsymbol{\psi}(\boldsymbol{x}) : \mathbb{R}^d \to \mathbb{R}^b$ is a non-negative basis function vector, and $\boldsymbol{\theta}$ $(\in \mathbb{R}^b)$ is a parameter vector. Since this model is the same form as that used in KLIEP for linear/kernel models (Section 2.3.2), we may use the same basis design idea described there.

For the above linear/kernel models, Eq.(21) is expressed as

$$\min_{\boldsymbol{\theta} \in \mathbb{R}^b} \left[\frac{1}{2} \boldsymbol{\theta}^\top \widehat{\boldsymbol{H}} \boldsymbol{\theta} - \widehat{\boldsymbol{h}}^\top \boldsymbol{\theta} \right],$$

where

$$\widehat{\boldsymbol{H}} := \frac{1}{n_{\mathrm{de}}} \sum_{j=1}^{n_{\mathrm{de}}} \boldsymbol{\psi}(\boldsymbol{x}_j^{\mathrm{de}}) \boldsymbol{\psi}(\boldsymbol{x}_j^{\mathrm{de}})^\top \quad \text{and} \quad \widehat{\boldsymbol{h}} := \frac{1}{n_{\mathrm{nu}}} \sum_{i=1}^{n_{\mathrm{nu}}} \boldsymbol{\psi}(\boldsymbol{x}_i^{\mathrm{nu}}). \tag{22}$$

2.4.2 Implementation with Non-Negativity Constraint

Here, we describe an implementation of LSIF *with* non-negativity constraint.

Let us impose non-negativity constraint $\boldsymbol{\theta} \geq \mathbf{0}_b$ since the density-ratio function is non-negative by definition. Let us further add the following regularization term to the objective function:

$$\mathbf{1}_b^\top \boldsymbol{\theta} = \|\boldsymbol{\theta}\|_1 := \sum_{\ell=1}^{b} |\theta_\ell|.$$

The term $\mathbf{1}_b^\top \boldsymbol{\theta}$ works as the ℓ_1-regularizer if it is combined with the non-negativity constraint. Then the optimization problem is expressed as follows.

$$\min_{\boldsymbol{\theta} \in \mathbb{R}^b} \left[\frac{1}{2} \boldsymbol{\theta}^\top \widehat{\boldsymbol{H}} \boldsymbol{\theta} - \widehat{\boldsymbol{h}}^\top \boldsymbol{\theta} + \lambda \mathbf{1}_b^\top \boldsymbol{\theta} \right] \quad \text{s.t.} \quad \boldsymbol{\theta} \geq \mathbf{0}_b,$$

where λ (≥ 0) is the regularization parameter. We refer to this method as *constrained LSIF* (cLSIF; Kanamori et al., 2009). The cLSIF optimization problem is a convex quadratic program, so the unique global optimal solution may be computed by a standard optimization software.

We can also use the ℓ_2-regularizer $\boldsymbol{\theta}^\top \boldsymbol{\theta}$, instead of the ℓ_1-regularizer $\mathbf{1}_b^\top \boldsymbol{\theta}$, without changing the computational property (i.e., the optimization problem is still a convex quadratic program). However, using the ℓ_1-regularizer would be more advantageous since the solution tends to be *sparse*, i.e., many parameters take exactly zero (Williams, 1995; Tibshirani, 1996; Chen et al., 1998). Furthermore, as shown in Kanamori et al. (2009), the use of the ℓ_1-regularizer allows one to compute the entire *regularization path* efficiently (Best, 1982; Efron et al., 2004; Hastie et al., 2004), which highly improves the computational cost in the model selection phase.

An R implementation of cLSIF is available from

http://www.math.cm.is.nagoya-u.ac.jp/˜kanamori/software/LSIF/

2.4.3 Implementation without Non-Negativity Constraint

Here, we describe another implementation of LSIF *without* the non-negativity constraint called *unconstrained LSIF* (uLSIF).

Without the non-negativity constraint, the linear regularizer $\mathbf{1}_b^\top \boldsymbol{\theta}$ used in cLSIF does not work as a regularizer. For this reason, a quadratic regularizer $\boldsymbol{\theta}^\top \boldsymbol{\theta}$ is adopted here. Then we have the following optimization problem.

$$\min_{\boldsymbol{\theta} \in \mathbb{R}^b} \left[\frac{1}{2} \boldsymbol{\theta}^\top \widehat{\boldsymbol{H}} \boldsymbol{\theta} - \widehat{\boldsymbol{h}}^\top \boldsymbol{\theta} + \frac{\lambda}{2} \boldsymbol{\theta}^\top \boldsymbol{\theta} \right]. \tag{23}$$

Eq.(23) is an unconstrained convex quadratic program, and the solution can be computed *analytically* by solving the following system of linear equations:

$$(\widehat{\boldsymbol{H}} + \lambda \boldsymbol{I}_b) \boldsymbol{\theta} = \widehat{\boldsymbol{h}},$$

where $\boldsymbol{I}_b$ is the b-dimensional identity matrix. The solution $\widehat{\boldsymbol{\theta}}$ of the above equation is given by

$$\widehat{\boldsymbol{\theta}} = (\widehat{\boldsymbol{H}} + \lambda \boldsymbol{I}_b)^{-1}\widehat{\boldsymbol{h}}.$$

Since the non-negativity constraint $\boldsymbol{\theta} \geq \mathbf{0}_b$ was dropped, some of the obtained parameters could be negative. To compensate for this approximation error, the solution may be modified as follows (Kanamori et al., 2012):

$$\max(0, \boldsymbol{\psi}(\boldsymbol{x})^\top \widehat{\boldsymbol{\theta}}).$$

This is the solution of the approximation method called *unconstrained LSIF* (uLSIF; Kanamori et al., 2009). An advantage of uLSIF is that the solution can be analytically computed just by solving a system of linear equations. Therefore, its computation is stable when λ is not too small.

A practically important advantage of uLSIF over cLSIF is that the score of *leave-one-out cross-validation* (LOOCV) can be computed analytically (Kanamori et al., 2009)—thanks to this property, the computational complexity for performing LOOCV is the same order as just computing a single solution.

A MATLAB® implementation of uLSIF is available from

http://sugiyama-www.cs.titech.ac.jp/˜sugi/software/uLSIF/

and an R implementation of uLSIF is available from

http://www.math.cm.is.nagoya-u.ac.jp/˜kanamori/software/LSIF/

2.4.4 Remarks

One can successfully avoid density estimation by least-squared density-ratio fitting. The least-squares methods for linear/kernel models are computationally more advantageous than alternative approaches such as moment matching (Section 2.1), probabilistic classification (Section 2.2), and density matching (Section 2.3). Indeed, the constrained method (cLSIF) for the ℓ_1-regularizer is equipped with a *regularization path tracking* algorithm. Furthermore, the unconstrained method (uLSIF) allows one to compute the density-ratio estimator analytically; the leave-one-out cross-validation score can also be computed in a closed form. Thus, the overall computation of uLSIF including model selection is highly efficient.

The fact that uLSIF has an analytic-form solution is actually very useful beyond its computational efficiency. When one wants to optimize some criterion defined using a density-ratio estimate (e.g., *mutual information*, see Cover and Thomas, 2006), the analytic-form solution of uLSIF allows one to compute the *derivative* of the target criterion analytically. Then one can develop, e.g., gradient-based and (quasi-)Newton algorithms for optimization. This property can be successfully utilized, e.g., in identifying the central subspace in *sufficient dimension reduction* (Suzuki and Sugiyama, 2010), finding independent components in *independent component analysis* (Suzuki and Sugiyama, 2011), performing dependence-minimizing regression in *causality learning* (Yamada and Sugiyama, 2010), and identifying the hetero-distributional subspace in *direct density-ratio estimation with dimensionality reduction* (Sugiyama et al., 2011b).

3 Unified Framework by Density-Ratio Matching

As reviewed in the previous section, various density-ratio estimation methods have been developed so far. In this section, we propose a new framework of density-ratio estimation by *density-ratio matching* under the *Bregman divergence* (Bregman, 1967), which includes various useful divergences (Banerjee et al., 2005; Stummer, 2007). This framework is a natural extension of the least-squares approach described in Section 2.4, and includes the existing approaches reviewed in the previous section as special cases (Section 3.2). Then we provide interpretation of density-ratio matching from two different views in Section 3.3. Finally, we give a new instance of density-ratio matching based on the *power divergence* in Section 3.4.

3.1 Basic Framework

A basic idea of density-ratio matching is to directly fit a density-ratio model $r(\boldsymbol{x})$ to the true density-ratio function $r^*(\boldsymbol{x})$ under some divergence. At a glance, this density-ratio matching problem is equivalent to the *regression problem*, which is aimed at estimating a real-valued function. However, density-ratio matching is essentially different from regression since samples of the true density-ratio function are not available. Here, we employ the *Bregman* (BR) divergence for measuring the discrepancy between the true density-ratio function and the density-ratio model.

The BR divergence is an extension of the *Euclidean distance* to a class of divergences that share similar properties. Let f be a differentiable and *strictly convex* function. Then the BR divergence associated with f from t^* to t is defined as

$$\mathrm{BR}'_f(t^*\|t) := f(t^*) - f(t) - \partial f(t)(t^* - t),$$

where ∂f is the derivative of f. Note that

$$f(t) + \partial f(t)(t^* - t)$$

is the value of the first-order *Taylor expansion* of f around t evaluated at t^*. Thus, the BR divergence evaluates the difference between the value of f at point t^* and its linear extrapolation from t (see Figure 1). $\mathrm{BR}'_f(t^*\|t)$ is a convex function with respect to t^*, but not necessarily convex with respect to t.

Here the discrepancy from the true density-ratio function r^* to a density-ratio model r is measured using the BR divergence as

$$\begin{aligned}\mathrm{BR}'_f\left(r^*\|r\right) := \int p^*_{\mathrm{de}}(\boldsymbol{x})\Big(&f(r^*(\boldsymbol{x})) - f(r(\boldsymbol{x})) \\ &- \partial f(r(\boldsymbol{x}))(r^*(\boldsymbol{x}) - r(\boldsymbol{x}))\Big)\mathrm{d}\boldsymbol{x}. \qquad (24)\end{aligned}$$

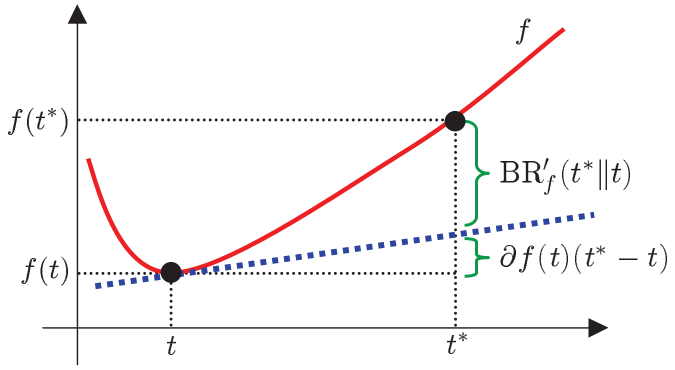


Figure 1: Bregman divergence $\mathrm{BR}'_f(t^*\|t)$.

A motivation for this choice is that the BR divergence allows one to directly obtain an *empirical approximation* for any f. Indeed, let us first extract a relevant part of $\mathrm{BR}'_f(r^*\|r)$ as

$$\mathrm{BR}'_f(r^*\|r) = \mathrm{BR}_f(r) + C,$$

where $C := \int p^*_{\mathrm{de}}(\boldsymbol{x})f(r^*(\boldsymbol{x}))\mathrm{d}\boldsymbol{x}$ is a constant independent of r, and

$$\mathrm{BR}_f(r) := \int p^*_{\mathrm{de}}(\boldsymbol{x})\Big(\partial f(r(\boldsymbol{x}))r(\boldsymbol{x}) - f(r(\boldsymbol{x}))\Big)\mathrm{d}\boldsymbol{x} - \int p^*_{\mathrm{nu}}(\boldsymbol{x})\partial f(r(\boldsymbol{x}))\mathrm{d}\boldsymbol{x}. \tag{25}$$

Then an empirical approximation $\widehat{\mathrm{BR}}_f(r)$ of $\mathrm{BR}_f(r)$ is given by

$$\widehat{\mathrm{BR}}_f(r) := \frac{1}{n_{\mathrm{de}}}\sum_{j=1}^{n_{\mathrm{de}}}\Big(\partial f(r(\boldsymbol{x}^{\mathrm{de}}_j))r(\boldsymbol{x}^{\mathrm{de}}_j) - f(r(\boldsymbol{x}^{\mathrm{de}}_j))\Big) - \frac{1}{n_{\mathrm{nu}}}\sum_{i=1}^{n_{\mathrm{nu}}}\partial f(r(\boldsymbol{x}^{\mathrm{nu}}_i)). \tag{26}$$

This immediately gives the following optimization criterion.

$$\min_r \widehat{\mathrm{BR}}_f(r),$$

where r is searched within some class of functions.

3.2 Existing Methods as Density-Ratio Matching

Here, we show that various density-ratio estimation methods reviewed in the previous section can be accommodated in the density-ratio matching framework (see Table 1).

3.2.1 Least-Squares Importance Fitting

Here, we show that the *least-squares importance fitting* (LSIF) approach introduced in Section 2.4.1 is an instance of density-ratio matching. More specifically, there exists a

Table 1: Summary of density-ratio estimation methods. In the table, 'LSIF', 'KMM', 'LR', and 'KLIEP' stand for 'least-squares importance fitting', 'kernel mean matching', 'logistic regression', and 'Kullback-Leibler Importance Estimation Procedure', respectively.

Method (Section)	$f(t)$	Model selection	Optimization
LSIF (3.2.1)	$(t-1)^2/2$	Available	Analytic
KMM (3.2.2)	$(t-1)^2/2$	Partially unavailable	Analytic
LR (3.2.3)	$t\log t-(1+t)\log(1+t)$	Available	Convex
KLIEP (3.2.4)	$t\log t - t$	Available	Convex
Robust (3.4)	$(t^{1+\alpha}-t)/\alpha$, $\alpha>0$	Available	Convex ($0<\alpha\leq 1$) Non-convex ($\alpha>1$)

BR divergence such that the optimization problem of density-ratio matching is reduced to that of LSIF.

When

$$f(t)=\frac{1}{2}(t-1)^2,$$

BR (24) is reduced to the squared (SQ) distance:

$$\mathrm{SQ}'(t^*\|t):=\frac{1}{2}(t^*-t)^2.$$

Following Eqs.(25) and (26), let us denote SQ without an irrelevant constant term by $\mathrm{SQ}\,(r)$ and its empirical approximation by $\widehat{\mathrm{SQ}}\,(r)$, respectively:

$$\mathrm{SQ}\,(r):=\frac{1}{2}\int p^*_{\mathrm{de}}(\boldsymbol{x})r(\boldsymbol{x})^2\mathrm{d}\boldsymbol{x}-\int p^*_{\mathrm{nu}}(\boldsymbol{x})r(\boldsymbol{x})\mathrm{d}\boldsymbol{x},$$

$$\widehat{\mathrm{SQ}}\,(r):=\frac{1}{2n_{\mathrm{de}}}\sum_{j=1}^{n_{\mathrm{de}}}r(\boldsymbol{x}_j^{\mathrm{de}})^2-\frac{1}{n_{\mathrm{nu}}}\sum_{i=1}^{n_{\mathrm{nu}}}r(\boldsymbol{x}_i^{\mathrm{nu}}).$$

This agrees with the LSIF formulation given in Section 2.4.1.

3.2.2 Kernel Mean Matching

Here, we show that the solution of the moment matching method, *kernel mean matching* (KMM) introduced in Section 2.1, actually agrees with that of *unconstrained LSIF* (uLSIF; see Section 2.4.3) for specific kernel models. Since uLSIF was shown to be an instance of density-ratio matching in Section 3.2.1, the KMM solution can also be obtained in the density-ratio matching framework.

Let us consider the following kernel density-ratio model:

$$r(\boldsymbol{x})=\sum_{\ell=1}^{n_{\mathrm{de}}}\theta_\ell K(\boldsymbol{x},\boldsymbol{x}_\ell^{\mathrm{de}}), \qquad (27)$$

where $K(\boldsymbol{x}, \boldsymbol{x}')$ is a *universal reproducing kernel* (Steinwart, 2001) such as the Gaussian kernel (7). Note that uLSIF and KLIEP use the numerator samples $\{\boldsymbol{x}_i^{\mathrm{nu}}\}_{i=1}^{n_{\mathrm{nu}}}$ as Gaussian centers, while the model (27) adopts the denominator samples $\{\boldsymbol{x}_j^{\mathrm{de}}\}_{j=1}^{n_{\mathrm{de}}}$ as Gaussian centers. For the density-ratio model (27), the matrix $\widehat{\boldsymbol{H}}$ and the vector $\widehat{\boldsymbol{h}}$ defined by Eq.(22) are expressed as

$$\widehat{\boldsymbol{H}} = \frac{1}{n_{\mathrm{de}}} \boldsymbol{K}_{\mathrm{de,de}}^2 \quad \text{and} \quad \widehat{\boldsymbol{h}} = \frac{1}{n_{\mathrm{nu}}} \boldsymbol{K}_{\mathrm{de,nu}} \mathbf{1}_{n_{\mathrm{nu}}},$$

where $\boldsymbol{K}_{\mathrm{de,de}}$ and $\boldsymbol{K}_{\mathrm{de,nu}}$ are defined in Eq.(8). Then the (unregularized) uLSIF solution (see Section 2.4.3 for details) is expressed as

$$\widehat{\boldsymbol{\theta}}_{\mathrm{uLSIF}} = \widehat{\boldsymbol{H}}^{-1} \widehat{\boldsymbol{h}} = \frac{n_{\mathrm{de}}}{n_{\mathrm{nu}}} \boldsymbol{K}_{\mathrm{de,de}}^{-2} \boldsymbol{K}_{\mathrm{de,nu}} \mathbf{1}_{n_{\mathrm{nu}}}. \tag{28}$$

On the other hand, let us consider an inductive variant of KMM for the kernel model (27) (see Section 2.1.2). For the density-ratio model (27), the design matrix $\boldsymbol{\Psi}_{\mathrm{de}}$ defined by Eq.(5) agrees with $\boldsymbol{K}_{\mathrm{de,de}}$. Then the KMM solution is given as follows (see Section 2.1.2):

$$\widehat{\boldsymbol{\theta}}_{\mathrm{KMM}} = \frac{n_{\mathrm{de}}}{n_{\mathrm{nu}}} (\boldsymbol{\Psi}_{\mathrm{de}} \boldsymbol{K}_{\mathrm{de,de}} \boldsymbol{\Psi}_{\mathrm{de}})^{-1} \boldsymbol{\Psi}_{\mathrm{de}} \boldsymbol{K}_{\mathrm{de,nu}} \mathbf{1}_{n_{\mathrm{nu}}} = \widehat{\boldsymbol{\theta}}_{\mathrm{uLSIF}}.$$

3.2.3 Logistic Regression

Here, we show that the *logistic regression* approach introduced in Section 2.2.2 is an instance of density-ratio matching. More specifically, there exists a BR divergence such that the optimization problem of density-ratio matching is reduced to that of the logistic regression approach.

When

$$f(t) = t \log t - (1+t) \log(1+t),$$

BR (24) is reduced to the *binary Kullback-Leibler* (BKL) divergence:

$$\mathrm{BKL}'(t^* \| t) := (1+t^*) \log \frac{1+t}{1+t^*} + t^* \log \frac{t}{t^*}.$$

The name 'BKL' comes from the fact that $\mathrm{BKL}'(t^* \| t)$ is expressed as

$$\mathrm{BKL}'(t^* \| t) = (1+t^*) \mathrm{KL}_{\mathrm{bin}} \left(\frac{1}{1+t^*} \middle\| \frac{1}{1+t} \right),$$

where $\mathrm{KL}_{\mathrm{bin}}$ is the KL divergence for binary random variables defined as

$$\mathrm{KL}_{\mathrm{bin}}(p, q) := p \log \frac{p}{q} + (1-p) \log \frac{1-p}{1-q}$$

for $0 < p, q < 1$. Thus, BKL' agrees with $\mathrm{KL}_{\mathrm{bin}}$ up to the constant factor $(1+t^*)$.

Following Eqs.(25) and (26), let us denote BKL without an irrelevant constant term by $\mathrm{BKL}\,(r)$ and its empirical approximation by $\widehat{\mathrm{BKL}}\,(r)$, respectively:

$$\mathrm{BKL}\,(r) := -\int p^*_{\mathrm{de}}(\boldsymbol{x}) \log \frac{1}{1+r(\boldsymbol{x})}\mathrm{d}\boldsymbol{x} - \int p^*_{\mathrm{nu}}(\boldsymbol{x}) \log \frac{r(\boldsymbol{x})}{1+r(\boldsymbol{x})}\mathrm{d}\boldsymbol{x},$$

$$\widehat{\mathrm{BKL}}\,(r) := -\frac{1}{n_{\mathrm{de}}}\sum_{j=1}^{n_{\mathrm{de}}} \log \frac{1}{1+r(\boldsymbol{x}^{\mathrm{de}}_j)} - \frac{1}{n_{\mathrm{nu}}}\sum_{i=1}^{n_{\mathrm{nu}}} \log \frac{r(\boldsymbol{x}^{\mathrm{nu}}_i)}{1+r(\boldsymbol{x}^{\mathrm{nu}}_i)}. \tag{29}$$

Eq.(29) is a generalized expression of logistic regression (Qin, 1998). Indeed, when $n_{\mathrm{de}} = n_{\mathrm{nu}}$, the ordinary logistic regression formulation (11) can be obtained from Eq.(29) (up to a regularizer) if the *log-linear* density-ratio model (18) without the constant term θ_0 is used.

3.2.4 Kullback-Leibler Importance Estimation Procedure

Here, we show that the *KL importance estimation procedure* (KLIEP) introduced in Section 2.3.1 is an instance of density-ratio matching. More specifically, there exists a BR divergence such that the optimization problem of density-ratio matching is reduced to that of the KLIEP approach.

When

$$f(t) = t\log t - t,$$

BR (24) is reduced to the *unnormalized Kullback-Leibler* (UKL) divergence:

$$\mathrm{UKL}'(t^*\|t) := t^*\log\frac{t^*}{t} - t^* + t.$$

Following Eqs.(25) and (26), let us denote UKL without an irrelevant constant term by $\mathrm{UKL}\,(r)$ and its empirical approximation by $\widehat{\mathrm{UKL}}\,(r)$, respectively:

$$\mathrm{UKL}\,(r) := \int p^*_{\mathrm{de}}(\boldsymbol{x}) r(\boldsymbol{x})\mathrm{d}\boldsymbol{x} - \int p^*_{\mathrm{nu}}(\boldsymbol{x}) \log r(\boldsymbol{x})\mathrm{d}\boldsymbol{x}, \tag{30}$$

$$\widehat{\mathrm{UKL}}\,(r) := \frac{1}{n_{\mathrm{de}}}\sum_{j=1}^{n_{\mathrm{de}}} r(\boldsymbol{x}^{\mathrm{de}}_j) - \frac{1}{n_{\mathrm{nu}}}\sum_{i=1}^{n_{\mathrm{nu}}} \log r(\boldsymbol{x}^{\mathrm{nu}}_i). \tag{31}$$

Let us further impose that the ratio model $r(\boldsymbol{x})$ is non-negative for all $\boldsymbol{x}$ and is normalized with respect to $\{\boldsymbol{x}^{\mathrm{de}}_j\}_{j=1}^{n_{\mathrm{de}}}$:

$$\frac{1}{n_{\mathrm{de}}}\sum_{j=1}^{n_{\mathrm{de}}} r(\boldsymbol{x}^{\mathrm{de}}_j) = 1.$$

Then the optimization criterion is reduced to as follows.

$$\begin{aligned}&\max_{r} \ \frac{1}{n_{\mathrm{nu}}}\sum_{i=1}^{n_{\mathrm{nu}}}\log r(\boldsymbol{x}_i^{\mathrm{nu}})\\ &\text{s.t.} \ \ \frac{1}{n_{\mathrm{de}}}\sum_{j=1}^{n_{\mathrm{de}}} r(\boldsymbol{x}_j^{\mathrm{de}}) = 1 \text{ and } r(\boldsymbol{x}) \geq 0 \text{ for all } \boldsymbol{x}.\end{aligned}$$

This agrees with the KLIEP formulation reviewed in Section 2.3.1.

3.3 Interpretation of Density-Ratio Matching

Here, we show the correspondence between the density-ratio matching approach and a divergence estimation method, and the correspondence between the density-ratio matching approach and a moment-matching approach.

3.3.1 Divergence Estimation View

We first show that our density-ratio matching formulation can be interpreted as *divergence estimation* based on the *Ali-Silvey-Csiszár* (ASC) divergence (Ali and Silvey, 1966; Csiszár, 1967), which is also known as the *f-divergence*.

Let us consider the ASC divergence for measuring the discrepancy between two probability density functions. An ASC divergence is defined using a *convex function* f such that $f(1) = 0$ as follows:

$$\mathrm{ASC}_f(p_{\mathrm{nu}}^* \| p_{\mathrm{de}}^*) := \int p_{\mathrm{de}}^*(\boldsymbol{x}) f\left(\frac{p_{\mathrm{nu}}^*(\boldsymbol{x})}{p_{\mathrm{de}}^*(\boldsymbol{x})}\right)\mathrm{d}\boldsymbol{x}. \qquad (32)$$

The ASC divergence is reduced to the *Kullback-Leibler* (KL) divergence (Kullback and Leibler, 1951) if $f(t) = t\log t$, and the *Pearson* (PE) divergence (Pearson, 1900) if $f(t) = \frac{1}{2}(t-1)^2$.

Let $\partial f(t)$ be the *sub-differential* of f at a point t ($\in \mathbb{R}$), which is a set defined as follows (Rockafellar, 1970):

$$\partial f(t) := \{z \in \mathbb{R} \mid f(s) \geq f(t) + z(s-t), \ \forall s \in \mathbb{R}\}.$$

If f is differentiable at t, then the sub-differential is reduced to the ordinary derivative. Although the sub-differential is a set in general, for simplicity, we treat $\partial f(r)$ as a single element if there is no confusion. Below, we assume that f is *closed*, i.e., its *epigraph* is a closed set (Rockafellar, 1970).

Let f^* be the *conjugate dual function* associated with f defined as

$$f^*(u) := \sup_t [tu - f(t)] = -\inf_t [f(t) - tu].$$

Since f is a closed convex function, we also have

$$f(t) = -\inf_{u}[f^*(u) - tu]. \tag{33}$$

For the KL divergence where $f(t) = t\log t$, the conjugate dual function is given by $f^*(u) = \exp(u-1)$. For the PE divergence where $f(t) = (t-1)^2/2$, the conjugate dual function is given by $f^*(u) = u^2/2 + u$.

Substituting Eq.(33) into Eq.(32), we have the following lower bound (Keziou, 2003):

$$\mathrm{ASC}_f(p^*_{\mathrm{nu}} \| p^*_{\mathrm{de}}) = -\inf_{g} \mathrm{ASC}'_f(g),$$

where

$$\mathrm{ASC}'_f(g) := \int f^*(g(\boldsymbol{x}))p^*_{\mathrm{de}}(\boldsymbol{x})\mathrm{d}\boldsymbol{x} - \int g(\boldsymbol{x})p^*_{\mathrm{nu}}(\boldsymbol{x})\mathrm{d}\boldsymbol{x}. \tag{34}$$

By taking the derivative of the integrand for each $\boldsymbol{x}$ and equating it to zero, we can show that the infimum of ASC'_f is attained at g such that

$$\partial f^*(g(\boldsymbol{x})) = \frac{p^*_{\mathrm{nu}}(\boldsymbol{x})}{p^*_{\mathrm{de}}(\boldsymbol{x})} = r^*(\boldsymbol{x}).$$

Thus, minimizing $\mathrm{ASC}'_f(g)$ yields the true density-ratio function $r^*(\boldsymbol{x})$.

For some g, there exists r such that

$$g = \partial f(r).$$

Then $f^*(g)$ is expressed as

$$f^*(g) = \sup_{s}\Big[s\partial f(r) - f(s)\Big].$$

According to the *variational principle* (Jordan et al., 1999), the supremum in the right-hand side of the above equation is attained at $s = r$. Thus, we have

$$f^*(g) = r\partial f(r) - f(r).$$

Then the lower bound $\mathrm{ASC}'_f(g)$ defined by Eq.(34) can be expressed as

$$\mathrm{ASC}'_f(g) = \int p^*_{\mathrm{de}}(\boldsymbol{x})\Big(r(\boldsymbol{x})\partial f(r(\boldsymbol{x})) - f(r(\boldsymbol{x}))\Big)\mathrm{d}\boldsymbol{x} - \int \partial f(r(\boldsymbol{x}))p^*_{\mathrm{nu}}(\boldsymbol{x})\mathrm{d}\boldsymbol{x}.$$

This is equivalent to the criterion BR_f defined by Eq.(25). Thus, density-ratio matching under the BR divergence can be interpreted as divergence estimation under the ASC divergence.

3.3.2 Moment Matching View

Next, we investigate the correspondence between the density-ratio matching approach and a moment-matching approach. To this end, we focus on the ideal situation where the true density-ratio function r^* is included in the density-ratio model r.

The non-linear version of finite-order moment matching (see Section 2.1.1) learns the density-ratio model r so that the following criterion is minimized:

$$\left\| \int \boldsymbol{\phi}(\boldsymbol{x}) r(\boldsymbol{x}) p^*_{\mathrm{de}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} - \int \boldsymbol{\phi}(\boldsymbol{x}) p^*_{\mathrm{nu}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \right\|^2,$$

where $\boldsymbol{\phi}(\boldsymbol{x}) : \mathbb{R}^d \to \mathbb{R}^m$ is some vector-valued function. Under the assumption that the density-ratio model r can represent the true density-ratio r^*, we have the following estimation equation:

$$\int \boldsymbol{\phi}(\boldsymbol{x}) r(\boldsymbol{x}) p^*_{\mathrm{de}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} - \int \boldsymbol{\phi}(\boldsymbol{x}) p^*_{\mathrm{nu}}(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} = \mathbf{0}_m, \tag{35}$$

where $\mathbf{0}_m$ denotes the m-dimensional vector with all zeros.

On the other hand, the density-ratio matching approach described in Section 3.1 learns the density-ratio model r so that the following criterion is minimized:

$$\int p^*_{\mathrm{de}}(\boldsymbol{x}) \partial f(r(\boldsymbol{x})) r(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} - \int p^*_{\mathrm{de}}(\boldsymbol{x}) f(r(\boldsymbol{x})) \mathrm{d}\boldsymbol{x} - \int p^*_{\mathrm{nu}}(\boldsymbol{x}) \partial f(r(\boldsymbol{x})) \mathrm{d}\boldsymbol{x}.$$

Taking the derivative of the above criterion with respect to parameters in the density-ratio model r and equate it to zero, we have the following estimation equation:

$$\int p^*_{\mathrm{de}}(\boldsymbol{x}) r(\boldsymbol{x}) \nabla r(\boldsymbol{x}) \partial^2 f(r(\boldsymbol{x})) \mathrm{d}\boldsymbol{x} - \int p^*_{\mathrm{nu}}(\boldsymbol{x}) \nabla r(\boldsymbol{x}) \partial^2 f(r(\boldsymbol{x})) \mathrm{d}\boldsymbol{x} = \mathbf{0}_b,$$

where ∇ denotes the differential operator with respect to parameters in the density-ratio model r, and b is the number of parameters. This implies that putting

$$\boldsymbol{\phi}(\boldsymbol{x}) = \nabla r(\boldsymbol{x}) \partial^2 f(r(\boldsymbol{x}))$$

in Eq.(35) gives the same estimation equation as density-ratio matching, resulting in the same optimal solution.

3.4 Basu's Power Divergence for Robust Density-Ratio Estimation

Finally, we introduce a new instance of density-ratio matching based on Basu's *power* divergence (BA divergence; Basu et al., 1998).

3.4.1 Derivation

For $\alpha > 0$, let

$$f(t) = \frac{t^{1+\alpha} - t}{\alpha}.$$

Then BR (24) is reduced to the BA divergence:

$$\mathrm{BA}'_\alpha(t^* \| t) := t^\alpha (t - t^*) - \frac{t^*(t^\alpha - (t^*)^\alpha)}{\alpha}.$$

Following Eqs.(25) and (26), let us denote BA'_α without an irrelevant constant term by $\mathrm{BA}_\alpha(r)$ and its empirical approximation by $\widehat{\mathrm{BA}}_\alpha(r)$, respectively:

$$\begin{aligned}
\mathrm{BA}_\alpha(r) &:= \int p^*_{\mathrm{de}}(\boldsymbol{x}) r(\boldsymbol{x})^{\alpha+1} \mathrm{d}\boldsymbol{x} - \left(1 + \frac{1}{\alpha}\right) \int p^*_{\mathrm{nu}}(\boldsymbol{x}) r(\boldsymbol{x})^\alpha \mathrm{d}\boldsymbol{x} + \frac{1}{\alpha}, \\
\widehat{\mathrm{BA}}_\alpha(r) &:= \frac{1}{n_{\mathrm{de}}} \sum_{j=1}^{n_{\mathrm{de}}} r(\boldsymbol{x}^{\mathrm{de}}_j)^{\alpha+1} - \left(1 + \frac{1}{\alpha}\right) \frac{1}{n_{\mathrm{nu}}} \sum_{i=1}^{n_{\mathrm{nu}}} r(\boldsymbol{x}^{\mathrm{nu}}_i)^\alpha + \frac{1}{\alpha}.
\end{aligned}$$

The density-ratio model r is determined so that $\widehat{\mathrm{BA}}_\alpha(r)$ is minimized.

When $\alpha = 1$, the BA divergence is reduced to the twice SQ divergence (see Section 2.4):

$$\widehat{\mathrm{BA}}_1 = 2\widehat{\mathrm{SQ}}.$$

Similarly, the fact

$$\lim_{\alpha \to 0} \frac{t^\alpha - 1}{\alpha} = \log t$$

implies that the BA divergence tends to the UKL divergence as $\alpha \to 0$ (see Section 3.2.4):

$$\lim_{\alpha \to 0} \widehat{\mathrm{BA}}_\alpha(r) = \frac{1}{n_{\mathrm{de}}} \sum_{j=1}^{n_{\mathrm{de}}} r(\boldsymbol{x}^{\mathrm{de}}_j) - \frac{1}{n_{\mathrm{nu}}} \sum_{i=1}^{n_{\mathrm{nu}}} \log r(\boldsymbol{x}^{\mathrm{nu}}_i) = \widehat{\mathrm{UKL}}(r).$$

Thus, the BA divergence essentially includes the SQ and UKL divergences as special cases, and is substantially more general.

3.4.2 Robustness

Let us take the derivative of $\widehat{\mathrm{BA}}_\alpha(r)$ with respect to parameters included in the density-ratio model r, and equate it to zero. Then we have the following estimation equation:

$$\frac{1}{n_{\mathrm{de}}} \sum_{j=1}^{n_{\mathrm{de}}} r(\boldsymbol{x}^{\mathrm{de}}_j)^\alpha \nabla r(\boldsymbol{x}^{\mathrm{de}}_j) - \frac{1}{n_{\mathrm{nu}}} \sum_{i=1}^{n_{\mathrm{nu}}} r(\boldsymbol{x}^{\mathrm{nu}}_i)^{\alpha-1} \nabla r(\boldsymbol{x}^{\mathrm{nu}}_i) = \boldsymbol{0}_b, \tag{36}$$

where ∇ is the differential operator with respect to parameters in the density-ratio model r, b denotes the number of parameters, and $\mathbf{0}_b$ denotes the b-dimensional vector with all zeros.

As explained in Section 3.4.1, the BA method with $\alpha \to 0$ corresponds to KLIEP (using the UKL divergence). According to Eq.(31), the estimation equation of KLIEP is given as follows (this also agrees with Eq.(36) with $\alpha = 0$):

$$\frac{1}{n_{\mathrm{de}}}\sum_{j=1}^{n_{\mathrm{de}}} \nabla r(\boldsymbol{x}_j^{\mathrm{de}}) - \frac{1}{n_{\mathrm{nu}}}\sum_{i=1}^{n_{\mathrm{nu}}} r(\boldsymbol{x}_i^{\mathrm{nu}})^{-1}\nabla r(\boldsymbol{x}_i^{\mathrm{nu}}) = \mathbf{0}_b.$$

Comparing this with Eq.(36), we see that the BA method can be regarded as a weighted version of KLIEP according to $r(\boldsymbol{x}_j^{\mathrm{de}})^\alpha$ and $r(\boldsymbol{x}_i^{\mathrm{nu}})^\alpha$. When $r(\boldsymbol{x}_j^{\mathrm{de}})$ and $r(\boldsymbol{x}_i^{\mathrm{nu}})$ are less than 1, the BA method down-weights the effect of those samples. Thus, 'outlying' samples relative to the density-ratio model r tend to have less influence on parameter estimation, which will lead to *robust* estimators (Basu et al., 1998).

Since LSIF corresponds to $\alpha = 1$, LSIF is more robust against outliers than KLIEP (which corresponds to $\alpha \to 0$) in the above sense, and BA with $\alpha > 1$ would be even more robust.

3.4.3 Numerical Examples

Here we illustrate the behavior of the robust density-ratio estimation method based on the BA divergence using artificial data sets.

Let the numerator and denominator densities be defined as follows (Figure 2(a)):

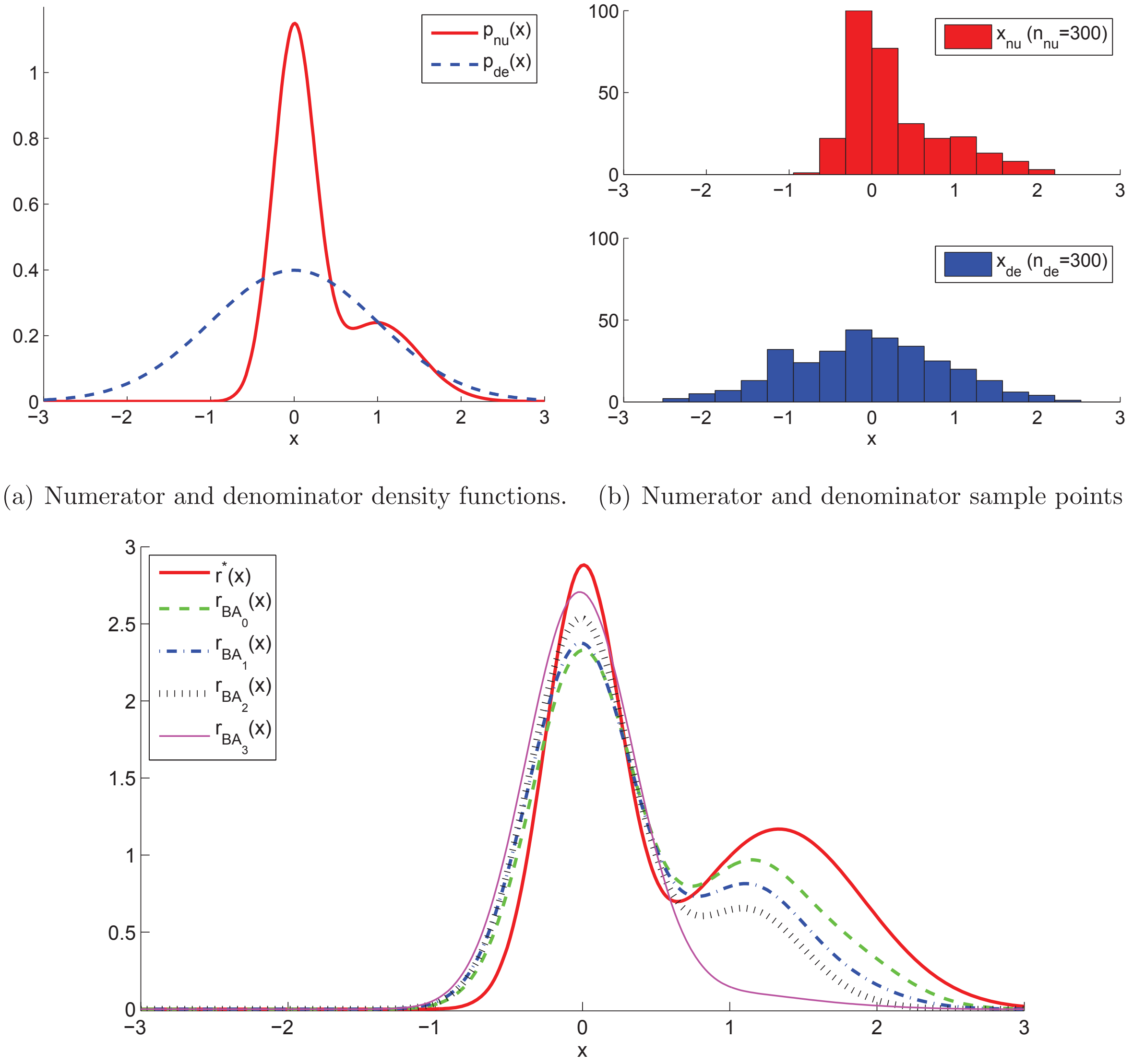
$$p^*_{\mathrm{nu}}(x) = 0.7N\left(x; 0, 0.25^2\right) + 0.3N\left(x; 1, 0.5^2\right) \quad \text{and} \quad p^*_{\mathrm{de}}(x) = N(x; 0, 1^2),$$

where $N(x; \mu, \sigma^2)$ denotes the Gaussian density with mean μ and variance σ^2,. We draw $n_{\mathrm{nu}} = n_{\mathrm{de}} = 300$ samples from each density, which are illustrated in Figure 2(b).

Here, we employ the Gaussian-kernel density-ratio model (17), and determine the model parameters so that $\widehat{\mathrm{BA}}_\alpha(r)$ with a quadratic regularizer is minimized under the non-negativity constraint:

$$\begin{aligned}
\min_{\boldsymbol{\theta}\in\mathbb{R}^b} \;& \Bigg[\frac{1}{n_{\mathrm{de}}}\sum_{j=1}^{n_{\mathrm{de}}}\left(\sum_{\ell=1}^{n_{\mathrm{nu}}}\theta_\ell K(\boldsymbol{x}_j^{\mathrm{nu}}, \boldsymbol{x}_\ell^{\mathrm{nu}})\right)^{\alpha+1} \\
&\quad - \left(1+\frac{1}{\alpha}\right)\frac{1}{n_{\mathrm{nu}}}\sum_{i=1}^{n_{\mathrm{nu}}}\left(\sum_{\ell=1}^{n_{\mathrm{nu}}}\theta_\ell K(\boldsymbol{x}_i^{\mathrm{de}}, \boldsymbol{x}_\ell^{\mathrm{nu}})\right)^{\alpha} + \lambda\boldsymbol{\theta}^\top\boldsymbol{\theta}\Bigg] \\
\text{s.t.} \;& \boldsymbol{\theta} \geq \mathbf{0}_b.
\end{aligned} \tag{37}$$

Note that this optimization problem is convex for $0 < \alpha \leq 1$. In our implementation, we solve the above optimization problem by gradient-projection, i.e., the parameters are iteratively updated by gradient descent with respect to the objective function, and the



(a) Numerator and denominator density functions.

(b) Numerator and denominator sample points

(c) True and estimated density-ratio functions

Figure 2: Numerical examples.

solution is projected back to the feasible region by rounding-up negative parameters to zero. Before solving the optimization problem (37), we run uLSIF (see Section 2.4.3) and obtain cross-validation estimates of the Gaussian width σ and the regularization parameter λ. We then fix the Gaussian width and the regularization parameter in the BA method to these values, and solve the optimization problem (37) by gradient-projection with $\boldsymbol{\theta} = \mathbf{1}_b/b$ as the initial solution.

Figure 2(c) shows the true and estimated density-ratio functions by the BA methods for $\alpha = 0, 1, 2, 3$. The true density-ratio function has two peaks—higher one at $x = 0$ and lower one at around $x = 1.2$. The graph shows that, as α increases, estimated density-ratio functions tend to focus on approximating the higher peak and ignore the lower peak. Thus, if numerator samples drawn from the right-hand Gaussian (i.e., $N\left(x; 1, 0.5^2\right)$) are regarded as outliers, the BA methods with larger α are more robust against these outliers.

We further investigate the issue of robustness against outliers more systematically. Let

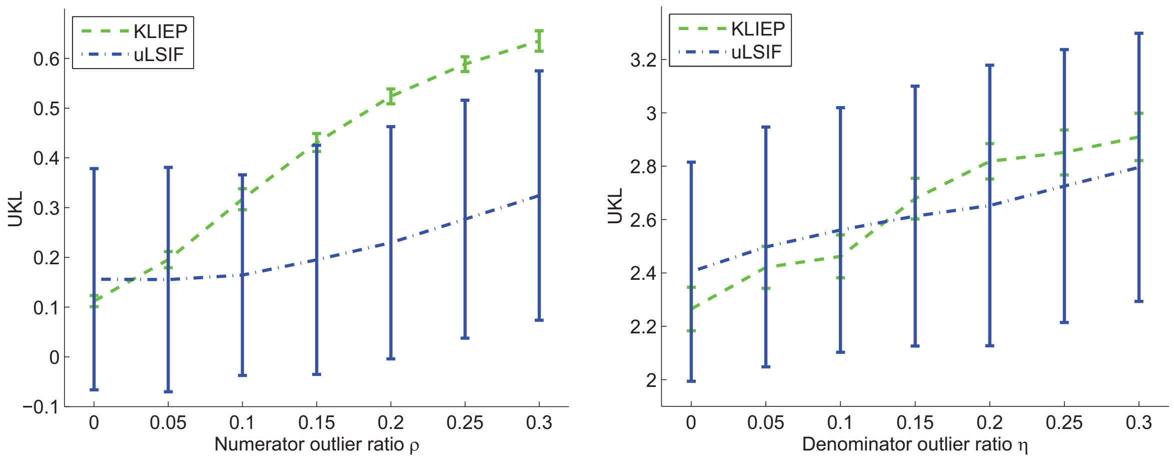
$$\begin{aligned} p^*_{\mathrm{nu}}(x) &= (1-\rho)N\left(x; 0, 0.25^2\right) + \rho N\left(x; 1, 0.5^2\right), \\ p^*_{\mathrm{de}}(x) &= (1-\eta)N(x; 0, 1^2) + \eta N(x; 0, 0.5^2), \end{aligned}$$

where ρ and η are the numerator and denominator outlier ratio, respectively; samples drawn from the second densities $N\left(x; 1, 0.5^2\right)$ and $N(x; 0, 0.5^2)$ are regarded as outliers. Let $n_{\mathrm{nu}} = n_{\mathrm{de}} = 300$, and we evaluate how the accuracy of density-ratio estimation is influenced by outliers. In the first set of experiments, we fix the denominator outlier ratio to $\eta = 0$ (i.e., no outlier) and change the numerator outlier ratio as $\rho = 0, 0.05, 0.1, \ldots, 0.3$. In the second set of experiments, we fix the numerator outlier ratio to $\rho = 0$ (i.e., no outlier) and change the denominator outlier ratio as $\eta = 0, 0.05, 0.1, \ldots, 0.3$. The approximation error of a density-ratio estimator $\widehat{r}$ is evaluated by $\mathrm{UKL}\left(\widehat{r}\right)$ defined by Eq.(30), which correspond to the BA divergence with $\alpha \to 0$ as explained in Section 3.4.1. Here, $\mathrm{UKL}\left(\widehat{r}\right)$ is numerically approximated using 1000 samples independently taken following $p^*_{\mathrm{nu}}(x)$ with $\rho = 0$ (i.e., no outliers) and 1000 samples independently taken following $p^*_{\mathrm{de}}(x)$ with $\eta = 0$ (i.e., no outliers). Note that these samples are not used for obtaining a density-ratio estimator $\widehat{r}$. For obtaining density-ratio estimators, we use off-the-shelf MATLAB implementation of KLIEP (which corresponds to the BA method with $\alpha \to 0$) and uLSIF (which corresponds to the BA method with $\alpha = 1$) available from the web (see Section 2.3 and Section 2.4). This renders a more practical setup of density-ratio estimation.

The median and standard deviation of UKL values for KLIEP and uLSIF over 100 runs are plotted in Figure 3. Note that the standard deviation is divided by 5 in the plots for clear visibility. The graphs show that KLIEP works better than uLSIF when the outlier ratio is small. This is natural consequences since KLIEP tries to minimizes UKL (see Section 3.2.4). However, as the outlier ratio increases, the approximation error of KLIEP grows rapidly. On the other hand, the approximation error of uLSIF grows rather mildly, showing the robustness of uLSIF against outliers. This phenomenon well agrees with the argument in Section 3.4.2.

However, the error bars of uLSIF are much larger than KLIEP. This would be caused by the fact that the ‘effective’ number of samples used in uLSIF is smaller than that of KLIEP due to the down-weighting effect explained in Section 3.4.2. Thus, the statistical efficiency of uLSIF would be lower than KLIEP, which is a common trade-off in robust statistical methods (Huber, 1981).

Another observation from these experimental results is that numerator outliers more strongly degrade the accuracy of KLIEP than denominator outliers.



(a) The numerator outlier ratio ρ is changed while the denominator outlier ratio is fixed to $\eta = 0$ (i.e., no outliers).

(b) The denominator outlier ratio η is changed while the numerator outlier ratio is fixed to $\rho = 0$ (i.e., no outliers).

Figure 3: The median and standard deviation of UKL values for KLIEP and uLSIF over 100 runs when the number of outlier samples is changed. For clear visibility, the standard deviation is divided by 5 in the plots.

4 Conclusions

In this paper, we addressed the problem of density-ratio estimation. We first provided a comprehensive review of density-ratio estimation methods, including the *moment matching approach* (Section 2.1), the *probabilistic classification approach* (Section 2.2), the *density matching approach* (Section 2.3), and the *density-ratio fitting approach* (Section 2.4). Then we proposed a novel framework of density-ratio estimation by density-ratio fitting under the *Bregman divergence* (Section 3.1). We showed that our novel framework accommodates the existing approaches reviewed above, and is substantially more general. Within this novel framework, we developed a robust density-ratio estimation method based on Basu's *power* divergence.

The power divergence method allows us to systematically compare the robustness of the density matching approach based on the KL divergence (KLIEP) and the density-ratio fitting approach based on the Pearson divergence (uLSIF). However, the robustness of the probabilistic classification approach is still unknown, which needs to be investigated in our future work.

Experimentally, we observed that numerator outliers tend to more significantly degrade the accuracy of KLIEP than denominator samples, while uLSIF is reasonably stable for both cases. It is interesting to investigate this experimental tendency theoretically, together with convergence properties of the robust method.

In the power divergence method, the choice of robustness parameter α is an open issue. Although there seems to be no universal way for this (Basu et al., 1998; Jones et al., 2001; Fujisawa and Eguchi, 2008), a practical approach would be to use cross-validation over a fixed divergence such as the squared distance.

Acknowledgements

Masashi Sugiyama was supported by SCAT, AOARD, and the JST PRESTO program. Taiji Suzuki was supported by MEXT Grant-in-Aid for Young Scientists (B) 22700289. Takafumi Kanamori was supported by MEXT Grant-in-Aid for Young Scientists (B) 20700251.

References

Ali SM, Silvey SD (1966) A general class of coefficients of divergence of one distribution from another. Journal of the Royal Statistical Society, Series B 28(1):131–142

Banerjee A, Merugu S, Dhillon IS, Ghosh J (2005) Clustering with Bregman divergences. Journal of Machine Learning Research 6:1705–1749

Basu A, Harris IR, Hjort NL, Jones MC (1998) Robust and efficient estimation by minimising a density power divergence. Biometrika 85(3):549–559

Best MJ (1982) An algorithm for the solution of the parametric quadratic programming problem. Tech. Rep. 82-24, Faculty of Mathematics, University of Waterloo

Bickel S, Bogojeska J, Lengauer T, Scheffer T (2008) Multi-task learning for HIV therapy screening. In: McCallum A, Roweis S (eds) Proceedings of 25th Annual International Conference on Machine Learning (ICML2008), pp 56–63

Bregman LM (1967) The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming. USSR Computational Mathematics and Mathematical Physics 7:200–217

Caruana R, Pratt L, Thrun S (1997) Multitask learning. Machine Learning 28:41–75

Cayton L (2008) Fast nearest neighbor retrieval for Bregman divergences. In: McCallum A, Roweis S (eds) Proceedings of the 25th Annual International Conference on Machine Learning (ICML2008), Omnipress, pp 112–119

Chen SS, Donoho DL, Saunders MA (1998) Atomic decomposition by basis pursuit. SIAM Journal on Scientific Computing 20(1):33–61

Cheng KF, Chu CK (2004) Semiparametric density estimation under a two-sample density ratio model. Bernoulli 10(4):583–604

Collins M, Schapire RE, Singer Y (2002) Logistic regression, adaboost and Bregman distances. Machine Learning 48(1-3):253–285

Cover TM, Thomas JA (2006) Elements of Information Theory, 2nd edn. John Wiley & Sons, Inc., Hoboken, NJ, USA

Csiszár I (1967) Information-type measures of difference of probability distributions and indirect observation. Studia Scientiarum Mathematicarum Hungarica 2:229–318

Dhillon I, Sra S (2006) Generalized nonnegative matrix approximations with Bregman divergences. In: Weiss Y, Schölkopf B, Platt J (eds) Advances in Neural Information Processing Systems 18, MIT Press, Cambridge, MA, pp 283–290

Efron B, Hastie T, Johnstone I, Tibshirani R (2004) Least angle regression. The Annals of Statistics 32(2):407–499

Fujisawa H, Eguchi S (2008) Robust parameter estimation with a small bias against heavy contamination. Journal of Multivariate Analysis 99(9):2053–2081

Gretton A, Smola A, Huang J, Schmittfull M, Borgwardt K, Schölkopf B (2009) Covariate shift by kernel mean matching. In: Quiñonero-Candela J, Sugiyama M, Schwaighofer A, Lawrence N (eds) Dataset Shift in Machine Learning, MIT Press, Cambridge, MA, USA, chap 8, pp 131–160

Hastie T, Tibshirani R, Friedman J (2001) The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer, New York, NY, USA

Hastie T, Rosset S, Tibshirani R, Zhu J (2004) The entire regularization path for the support vector machine. Journal of Machine Learning Research 5:1391–1415

Hido S, Tsuboi Y, Kashima H, Sugiyama M, Kanamori T (2008) Inlier-based outlier detection via direct density ratio estimation. In: Giannotti F, Gunopulos D, Turini F, Zaniolo C, Ramakrishnan N, Wu X (eds) Proceedings of IEEE International Conference on Data Mining (ICDM2008), Pisa, Italy, pp 223–232

Hido S, Tsuboi Y, Kashima H, Sugiyama M, Kanamori T (2011) Statistical outlier detection using direct density ratio estimation. Knowledge and Information Systems 26(2):309–336

Huang J, Smola A, Gretton A, Borgwardt KM, Schölkopf B (2007) Correcting sample selection bias by unlabeled data. In: Schölkopf B, Platt J, Hoffman T (eds) Advances in Neural Information Processing Systems 19, MIT Press, Cambridge, MA, USA, pp 601–608

Huber PJ (1981) Robust Statistics. Wiley, New York, NY, USA

Jones MC, Hjort NL, Harris IR, Basu A (2001) A comparison of related density-based minimum divergence estimators. Biometrika 88:865–873

Jordan MI, Ghahramani Z, Jaakkola TS, Saul LK (1999) An introduction to variational methods for graphical models. Machine Learning 37(2):183

Kanamori T, Hido S, Sugiyama M (2009) A least-squares approach to direct importance estimation. Journal of Machine Learning Research 10:1391–1445

Kanamori T, Suzuki T, Sugiyama M (2010) Theoretical analysis of density ratio estimation. IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences E93-A(4):787–798

Kanamori T, Suzuki T, Sugiyama M (2012) Kernel-based least-squares density-ratio estimation I: Statistical analysis. Machine Learning To appear

Kawahara Y, Sugiyama M (2009) Change-point detection in time-series data by direct density-ratio estimation. In: Park H, Parthasarathy S, Liu H, Obradovic Z (eds) Proceedings of 2009 SIAM International Conference on Data Mining (SDM2009), Sparks, Nevada, USA, pp 389–400

Keziou A (2003) Dual representation of ϕ-divergences and applications. Comptes Rendus Mathématique 336(10):857–862

Keziou A, Leoni-Aubin S (2005) Test of homogeneity in semiparametric two-sample density ratio models. Comptes Rendus Mathématique 340(12):905–910

Kimura M, Sugiyama M (2011) Dependence-maximization clustering with least-squares mutual information. Journal of Advanced Computational Intelligence and Intelligent Informatics 15(7):800–805

Kullback S, Leibler RA (1951) On information and sufficiency. Annals of Mathematical Statistics 22:79–86

Minka TP (2007) A comparison of numerical optimizers for logistic regression. Tech. rep., Microsoft Research, URL http://research.microsoft.com/˜minka/papers/logreg/minka-logreg.pdf

Murata N, Takenouchi T, Kanamori T, Eguchi S (2004) Information geometry of U-boost and Bregman divergence. Neural Computation 16(7):1437–1481

Nguyen X, Wainwright MJ, Jordan MI (2010) Estimating divergence functionals and the likelihood ratio by convex risk minimization. IEEE Transactions on Information Theory 56(11):5847–5861

Pearson K (1900) On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling. Philosophical Magazine Series 5 50(302):157–175

Qin J (1998) Inferences for case-control and semiparametric two-sample density ratio models. Biometrika 85(3):619–630

Quiñonero-Candela J, Sugiyama M, Schwaighofer A, Lawrence N (eds) (2009) Dataset Shift in Machine Learning. MIT Press, Cambridge, MA, USA

Rockafellar RT (1970) Convex Analysis. Princeton University Press, Princeton, NJ, USA

Schölkopf B, Smola AJ (2002) Learning with Kernels. MIT Press, Cambridge, MA, USA

Shimodaira H (2000) Improving predictive inference under covariate shift by weighting the log-likelihood function. Journal of Statistical Planning and Inference 90(2):227–244

Silverman BW (1978) Density ratios, empirical likelihood and cot death. Journal of the Royal Statistical Society, Series C 27(1):26–33

Smola A, Song L, Teo CH (2009) Relative novelty detection. In: van Dyk D, Welling M (eds) Proceedings of Twelfth International Conference on Artificial Intelligence and Statistics (AISTATS2009), Clearwater Beach, FL, USA, JMLR Workshop and Conference Proceedings, vol 5, pp 536–543

Steinwart I (2001) On the influence of the kernel on the consistency of support vector machines. Journal of Machine Learning Research 2:67–93

Stummer W (2007) Some Bregman distances between financial diffusion processes. Proceedings in Applied Mathematics and Mechanics 7:1050,503–1050,504

Sugiyama M (2010) Superfast-trainable multi-class probabilistic classifier by least-squares posterior fitting. IEICE Transactions on Information and Systems E93-D(10):2690–2701

Sugiyama M, Kawanabe M (2011) Machine Learning in Non-Stationary Environments: Introduction to Covariate Shift Adaptation. MIT Press, Cambridge, MA, USA, to appear

Sugiyama M, Müller KR (2005) Input-dependent estimation of generalization error under covariate shift. Statistics & Decisions 23(4):249–279

Sugiyama M, Krauledat M, Müller KR (2007) Covariate shift adaptation by importance weighted cross validation. Journal of Machine Learning Research 8:985–1005

Sugiyama M, Suzuki T, Nakajima S, Kashima H, von Bünau P, Kawanabe M (2008) Direct importance estimation for covariate shift adaptation. Annals of the Institute of Statistical Mathematics 60(4):699–746

Sugiyama M, Kanamori T, Suzuki T, Hido S, Sese J, Takeuchi I, Wang L (2009) A density-ratio framework for statistical data processing. IPSJ Transactions on Computer Vision and Applications 1:183–208

Sugiyama M, Takeuchi I, Suzuki T, Kanamori T, Hachiya H, Okanohara D (2010) Least-squares conditional density estimation. IEICE Transactions on Information and Systems E93-D(3):583–594

Sugiyama M, Suzuki T, Itoh Y, Kanamori T, Kimura M (2011a) Least-squares two-sample test. Neural Networks 24(7):735–751

Sugiyama M, Yamada M, von Bünau P, Suzuki T, Kanamori T, Kawanabe M (2011b) Direct density-ratio estimation with dimensionality reduction via least-squares hetero-distributional subspace search. Neural Networks 24(2):183–198

Sugiyama M, Suzuki T, Kanamori T (2012) Density Ratio Estimation in Machine Learning. Cambridge University Press, Cambridge, UK, to appear

Suzuki T, Sugiyama M (2009) Estimating squared-loss mutual information for independent component analysis. In: Adali T, Jutten C, Romano JMT, Barros AK (eds) Independent Component Analysis and Signal Separation, Springer, Berlin, Germany, Lecture Notes in Computer Science, vol 5441, pp 130–137

Suzuki T, Sugiyama M (2010) Sufficient dimension reduction via squared-loss mutual information estimation. In: Teh YW, Tiggerington M (eds) Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics (AISTATS2010), Sardinia, Italy, JMLR Workshop and Conference Proceedings, vol 9, pp 804–811

Suzuki T, Sugiyama M (2011) Least-squares independent component analysis. Neural Computation 23(1):284–301

Suzuki T, Sugiyama M, Sese J, Kanamori T (2008) Approximating mutual information by maximum likelihood density ratio estimation. In: Saeys Y, Liu H, Inza I, Wehenkel L, de Peer YV (eds) Proceedings of ECML-PKDD2008 Workshop on New Challenges for Feature Selection in Data Mining and Knowledge Discovery 2008 (FSDM2008), Antwerp, Belgium, JMLR Workshop and Conference Proceedings, vol 4, pp 5–20

Suzuki T, Sugiyama M, Kanamori T, Sese J (2009a) Mutual information estimation reveals global associations between stimuli and biological processes. BMC Bioinformatics 10(1):S52

Suzuki T, Sugiyama M, Tanaka T (2009b) Mutual information approximation via maximum likelihood estimation of density ratio. In: Proceedings of 2009 IEEE International Symposium on Information Theory (ISIT2009), Seoul, Korea, pp 463–467

Tibshirani R (1996) Regression shrinkage and subset selection with the lasso. Journal of the Royal Statistical Society, Series B 58(1):267–288

Tipping ME, Bishop CM (1999) Mixtures of probabilistic principal component analyzers. Neural Computation 11(2):443–482

Tsuboi Y, Kashima H, Hido S, Bickel S, Sugiyama M (2009) Direct density ratio estimation for large-scale covariate shift adaptation. Journal of Information Processing 17:138–155

Tsuda K, Rätsch G, Warmuth M (2005) Matrix exponential gradient updates for on-line learning and Bregman projection. In: Saul LK, Weiss Y, Bottou L (eds) Advances in Neural Information Processing Systems 17, MIT Press, Cambridge, MA, pp 1425–1432

Williams PM (1995) Bayesian regularization and pruning using a Laplace prior. Neural Computation 7(1):117–143

Wu L, Jin R, Hoi SCH, Zhu J, Yu N (2009) Learning Bregman distance functions and its application for semi-supervised clustering. In: Bengio Y, Schuurmans D, Lafferty J, Williams CKI, Culotta A (eds) Advances in Neural Information Processing Systems 22, pp 2089–2097

Yamada M, Sugiyama M (2009) Direct importance estimation with Gaussian mixture models. IEICE Transactions on Information and Systems E92-D(10):2159–2162

Yamada M, Sugiyama M (2010) Dependence minimizing regression with model selection for non-linear causal inference under non-Gaussian noise. In: Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence (AAAI2010), The AAAI Press, Atlanta, Georgia, USA, pp 643–648

Yamada M, Sugiyama M (2011) Cross-domain object matching with model selection. In: Gordon G, Dunson D, Dudík M (eds) Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics (AISTATS2011), Fort Lauderdale, Florida, USA, pp 807–815

Yamada M, Sugiyama M, Wichern G, Simm J (2010) Direct importance estimation with a mixture of probabilistic principal component analyzers. IEICE Transactions on Information and Systems E93-D(10):2846–2849

Yamada M, Sugiyama M, Wichern G, Simm J (2011) Improving the accuracy of least-squares probabilistic classifiers. IEICE Transactions on Information and Systems E94-D(6):1337–1340

Entropy **2013**, *15*, 80-112; doi:10.3390/e15010080

OPEN ACCESS

entropy

ISSN 1099-4300

www.mdpi.com/journal/entropy

Review

Machine Learning with Squared-Loss Mutual Information

Masashi Sugiyama

Department of Computer Science, Tokyo Institute of Technology 2-12-1 O-okayama, Meguro-ku, Tokyo 152-8552, Japan; E-Mail: sugi@cs.titech.ac.jp

Received: 29 October 2012; in revised form: 7 December 2012 / Accepted: 21 December 2012 / Published: 27 December 2012

Abstract: Mutual information (MI) is useful for detecting statistical independence between random variables, and it has been successfully applied to solving various machine learning problems. Recently, an alternative to MI called *squared-loss MI* (SMI) was introduced. While ordinary MI is the Kullback–Leibler divergence from the joint distribution to the product of the marginal distributions, SMI is its Pearson divergence variant. Because both the divergences belong to the f-divergence family, they share similar theoretical properties. However, a notable advantage of SMI is that it can be approximated from data in a computationally more efficient and numerically more stable way than ordinary MI. In this article, we review recent development in SMI approximation based on direct density-ratio estimation and SMI-based machine learning techniques such as independence testing, dimensionality reduction, canonical dependency analysis, independent component analysis, object matching, clustering, and causal inference.

Keywords: squared-loss mutual information; Pearson divergence; density-ratio estimation; independence testing; dimensionality reduction; independent component analysis; object matching; clustering; causal inference; machine learning

1. Introduction

Mutual information (MI) [1,2] for random variables $\boldsymbol{X}$ and $\boldsymbol{Y}$ is defined as:

$$\mathrm{MI}(\boldsymbol{X}, \boldsymbol{Y}) := \iint p(\boldsymbol{x}, \boldsymbol{y}) \log \frac{p(\boldsymbol{x}, \boldsymbol{y})}{p(\boldsymbol{x})p(\boldsymbol{y})} \mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{y}$$

where $p(\boldsymbol{x}, \boldsymbol{y})$ is the joint probability density of $\boldsymbol{X}$ and $\boldsymbol{Y}$, and $p(\boldsymbol{x})$ and $p(\boldsymbol{y})$ are the marginal probability densities of $\boldsymbol{X}$ and $\boldsymbol{Y}$, respectively (Precisely, $p(\boldsymbol{x}, \boldsymbol{y})$, $p(\boldsymbol{x})$, and $p(\boldsymbol{y})$ are different functions and thus should be denoted, e.g., by $p_{\mathbf{X},\mathbf{Y}}(\boldsymbol{x},\boldsymbol{y})$, $p_{\mathbf{X}}(\boldsymbol{x})$, and $p_{\mathbf{Y}}(\boldsymbol{y})$, respectively. However, we use the simplified notations for the sake of brevity). Statistically, MI can be regarded as the Kullback–Leibler divergence [3] from the joint density $p(\boldsymbol{x},\boldsymbol{y})$ to the product of the marginals $p(\boldsymbol{x})p(\boldsymbol{y})$, and thus can be regarded as a measure of statistical dependency between $\boldsymbol{X}$ and $\boldsymbol{Y}$. Estimation of MI from data samples has been one of the major challenges in information science and various approaches have been developed thus far.

The most naive approach to approximating MI from data would be to use a non-parametric density estimator such as kernel density estimation (KDE) [4], *i.e.*, the densities $p(\boldsymbol{x},\boldsymbol{y})$, $p(\boldsymbol{x})$, and $p(\boldsymbol{y})$ included in MI are separately estimated from samples, and the estimated densities are used for approximating MI. However, density estimation is known to be a hard problem [5] and division by estimated densities tends to magnify the estimation error. Therefore, the KDE-based MI approximator may not be reliable in practice.

Another approach uses histogram-based density estimators with data-dependent partitioning. In the context of estimating the Kullback–Leibler divergence [3], histogram-based methods have been studied thoroughly and their consistency has been established [6–8]. However, the rate of convergence has not been elucidated yet, and such histogram-based methods are strongly influenced by the curse of dimensionality. Thus, these methods may not be reliable in high-dimensional problems.

MI can be expressed in terms of the entropies as:

$$\mathrm{MI}(\boldsymbol{X},\boldsymbol{Y}) = H(\boldsymbol{X}) + H(\boldsymbol{Y}) - H(\boldsymbol{X},\boldsymbol{Y})$$

where $H(\boldsymbol{X})$ denotes the entropy of $\boldsymbol{X}$:

$$H(\boldsymbol{X}) := -\int p(\boldsymbol{x}) \log p(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}$$

Based on this expression, the nearest neighbor distance has been used for approximating MI [9]. Such a nearest neighbor approach was shown to perform better than the naive KDE-based approach [10], given that the number k of nearest neighbors is chosen appropriately—a small (large) k yields an estimator with small (large) bias and large (small) variance. However, appropriately determining the value of k so that the bias-variance trade-off is optimally controlled is not straightforward in the context of MI estimation. A similar nearest-neighbor idea has been applied to Kullback–Leibler divergence estimation [11], whose consistency has been proved for finite k—this is an interesting result since Kullback–Leibler divergence estimation is consistent even when density estimation is not consistent. However, the rate of convergence seems to be still an open research issue.

Approximation of the entropies based on the Edgeworth expansion has also been explored in the context of MI estimation [12]. Such a method works well when the target density is close to Gaussian. However, if the target density is far from Gaussian, the Edgeworth expansion method is no longer reliable.

More recently, an MI approximator via direct estimation of the density ratio $\frac{p(\boldsymbol{x},\boldsymbol{y})}{p(\boldsymbol{x})p(\boldsymbol{y})}$ has been developed [13], which is based on a Kullback–Leibler divergence approximator via direct density-ratio estimation [14–16]. The MI approximator is given as the solution of a convex optimization problem, which tends to be sparse [14]. A notable advantage of this density-ratio method is that it does not involve separate estimation of densities $p(\boldsymbol{x},\boldsymbol{y})$, $p(\boldsymbol{x})$, and $p(\boldsymbol{y})$, and it was proved to achieve the optimal non-parametric convergence rate. However, due to the "log" operation included in MI, this MI approximator is computationally rather expensive and susceptible to outliers [17,18].

To cope with these problems, a variant of MI called the *squared-loss mutual information* (SMI) [19] has been explored recently. SMI for $\boldsymbol{X}$ and $\boldsymbol{Y}$ is defined as:

$$\mathrm{SMI}(\boldsymbol{X}, \boldsymbol{Y}) := \frac{1}{2} \iint p(\boldsymbol{x})p(\boldsymbol{y}) \left(\frac{p(\boldsymbol{x}, \boldsymbol{y})}{p(\boldsymbol{x})p(\boldsymbol{y})} - 1 \right)^2 \mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{y}$$

SMI is the Pearson divergence [20] from the joint density $p(\boldsymbol{x}, \boldsymbol{y})$ to the product of the marginals $p(\boldsymbol{x})p(\boldsymbol{y})$. It is always non-negative and it vanishes if and only if $\boldsymbol{X}$ and $\boldsymbol{Y}$ are statistically independent. Note that both the Pearson divergence and the Kullback–Leibler divergence belong to the class of Ali–Silvey–Csiszár divergences (which is also known as f-divergences) [21,22], meaning that they share similar properties.

In a similar way to ordinary MI, SMI can be approximated accurately via direct estimation of the density ratio $\frac{p(\boldsymbol{x},\boldsymbol{y})}{p(\boldsymbol{x})p(\boldsymbol{y})}$ [19], which is based on a Pearson divergence approximator via direct density-ratio estimation [16,23]. This SMI approximator has various desirable properties: For example, it was proved to achieve the optimal non-parametric convergence rate [24], its solution can be obtained *analytically* just by solving a system of linear equations, it has superior numerical properties [25], and it is robust against outliers [17,18].

In particular, the property of the SMI approximator that an analytic solution is available is highly useful in machine learning, because this allows explicit computation of the *derivative* of the SMI approximator with respect to another parameter. For example, in supervised dimensionality reduction, linear transformation $\boldsymbol{U}$ for input $\boldsymbol{x}$ is optimized so that the transformed input $\boldsymbol{U}\boldsymbol{x}$ has the highest dependency on output $\boldsymbol{y}$. In this context, the derivative of the SMI estimator between $\boldsymbol{U}\boldsymbol{x}$ and $\boldsymbol{y}$ with respect to transformation $\boldsymbol{U}$ can be exploited for optimizing transformation $\boldsymbol{U}$. On the other hand, such derivative computation is not straightforward for the MI estimator whose solution is obtained via numerical optimization.

The purpose of this article is to review recent development in SMI approximation based on direct density-ratio estimation and SMI-based machine learning techniques. The remainder of this paper is structured as follows. After reviewing the SMI approximator based on direct density-ratio estimation in Section 2, we illustrate in Section 3 how the SMI approximator can be utilized for solving various machine learning tasks such as: independence testing [26], feature selection [19,27], feature extraction [28,29], canonical dependency analysis [30], independent component analysis [31], object matching [32], clustering [33,34], and causality learning [35].

2. Definition and Estimation of SMI

In this section, we review the definition of SMI and its approximator based on direct density-ratio estimation.

2.1. Definition of SMI

Let us consider two random variables $\boldsymbol{X} \in \mathcal{X}$ and $\boldsymbol{Y} \in \mathcal{Y}$, where $\mathcal{X}$ and $\mathcal{Y}$ are domains of $\boldsymbol{X}$ and $\boldsymbol{Y}$, respectively. Let $p(\boldsymbol{x}, \boldsymbol{y})$ be the joint probability density of $\boldsymbol{X}$ and $\boldsymbol{Y}$, and $p(\boldsymbol{x})$ and $p(\boldsymbol{y})$ be the marginal probability densities of $\boldsymbol{X}$ and $\boldsymbol{Y}$, respectively. The *squared-loss mutual information* (SMI) [19] for $\boldsymbol{X}$ and $\boldsymbol{Y}$ is defined as:

$$\mathrm{SMI}(\boldsymbol{X}, \boldsymbol{Y}) := \frac{1}{2} \iint p(\boldsymbol{x}) p(\boldsymbol{y}) \left(\frac{p(\boldsymbol{x}, \boldsymbol{y})}{p(\boldsymbol{x}) p(\boldsymbol{y})} - 1 \right)^2 \mathrm{d}\boldsymbol{x} \mathrm{d}\boldsymbol{y} \tag{1}$$

SMI is always non-negative and it takes zero if and only if $\boldsymbol{X}$ and $\boldsymbol{Y}$ are statistically independent. Hence, SMI can be used for detecting statistical independence between $\boldsymbol{X}$ and $\boldsymbol{Y}$.

Below, we consider the problem of estimating SMI from paired samples $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n$ drawn independently from the joint distribution with density $p(\boldsymbol{x}, \boldsymbol{y})$.

2.2. Least-Squares Estimation of SMI

Here, we review the basic idea and theoretical properties of the SMI approximator called *least-squares mutual information* (LSMI) [19].

2.2.1. SMI Approximation via Direct Density-Ratio Estimation

The basic idea of LSMI is to directly estimate the following *density-ratio* function without going through density estimation of $p(\boldsymbol{x}, \boldsymbol{y})$, $p(\boldsymbol{x})$, and $p(\boldsymbol{y})$:

$$r(\boldsymbol{x}, \boldsymbol{y}) := \frac{p(\boldsymbol{x}, \boldsymbol{y})}{p(\boldsymbol{x}) p(\boldsymbol{y})} \tag{2}$$

Let $g(\boldsymbol{x}, \boldsymbol{y})$ be a model of the density ratio $r(\boldsymbol{x}, \boldsymbol{y})$. In LSMI, the model is learned so that the following squared-error J is minimized:

$$\begin{aligned} J(g) &:= \frac{1}{2} \iint \left(g(\boldsymbol{x}, \boldsymbol{y}) - r(\boldsymbol{x}, \boldsymbol{y}) \right)^2 p(\boldsymbol{x}) p(\boldsymbol{y}) \mathrm{d}\boldsymbol{x} \mathrm{d}\boldsymbol{y} \\ &= \frac{1}{2} \iint g(\boldsymbol{x}, \boldsymbol{y})^2 p(\boldsymbol{x}) p(\boldsymbol{y}) \mathrm{d}\boldsymbol{x} \mathrm{d}\boldsymbol{y} - \iint g(\boldsymbol{x}, \boldsymbol{y}) p(\boldsymbol{x}, \boldsymbol{y}) \mathrm{d}\boldsymbol{x} \mathrm{d}\boldsymbol{y} + C \end{aligned} \tag{3}$$

where C is a constant defined by:

$$C := \frac{1}{2} \iint r(\boldsymbol{x}, \boldsymbol{y}) p(\boldsymbol{x}, \boldsymbol{y}) \mathrm{d}\boldsymbol{x} \mathrm{d}\boldsymbol{y}$$

Since J contains the expectations over unknown densities $p(\boldsymbol{x}) p(\boldsymbol{y})$ and $p(\boldsymbol{x}, \boldsymbol{y})$, the expectations are approximated by empirical averages. Then the LSMI optimization problem is given as follows:

$$\widehat{g} := \underset{g \in \mathcal{G}}{\operatorname{argmin}} \left[\frac{1}{2n^2} \sum_{i,j=1}^{n} g(\boldsymbol{x}_i, \boldsymbol{y}_j)^2 - \frac{1}{n} \sum_{i=1}^{n} g(\boldsymbol{x}_i, \boldsymbol{y}_i) \right] \tag{4}$$

where $\mathcal{G}$ is a function space from which g is searched.

Finally, the SMI approximator called LSMI is given as:

$$\mathrm{LSMI}(\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n) := \frac{1}{2n} \sum_{i=1}^n \widehat{g}(\boldsymbol{x}_i, \boldsymbol{y}_i) - \frac{1}{2} \tag{5}$$

or

$$\mathrm{LSMI}'(\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n) := -\frac{1}{2n^2} \sum_{i,j=1}^n \widehat{g}(\boldsymbol{x}_i, \boldsymbol{y}_j)^2 + \frac{1}{n} \sum_{i=1}^n \widehat{g}(\boldsymbol{x}_i, \boldsymbol{y}_i) - \frac{1}{2} \tag{6}$$

Equation (5) would be the simplest SMI approximator, while Equation (6) is suitable for theoretical analysis because this corresponds to the negative of the objective function (4) up to the constant $1/2$. These estimators are derived based on the following equivalent expressions of SMI:

$$\begin{aligned} \mathrm{SMI}(\boldsymbol{X}, \boldsymbol{Y}) &= \frac{1}{2} \iint r(\boldsymbol{x}, \boldsymbol{y}) p(\boldsymbol{x}, \boldsymbol{y}) \mathrm{d}\boldsymbol{x} \mathrm{d}\boldsymbol{y} - \frac{1}{2} && (7) \\ &= -\frac{1}{2} \iint r(\boldsymbol{x}, \boldsymbol{y})^2 p(\boldsymbol{x}) p(\boldsymbol{y}) \mathrm{d}\boldsymbol{x} \mathrm{d}\boldsymbol{y} + \iint r(\boldsymbol{x}, \boldsymbol{y}) p(\boldsymbol{x}, \boldsymbol{y}) \mathrm{d}\boldsymbol{x} \mathrm{d}\boldsymbol{y} - \frac{1}{2} && (8) \end{aligned}$$

Equation (7) is obtained by expanding the squared term in Equation (1), applying Equation (2) to the squared density-ratio term once, and showing that the cross-term and the remaining terms are -1 and $1/2$, respectively. Equivalence between Equations (7) and (8) can be confirmed by applying Equation (2) to the first term in Equation (8) once. Note that Equation (8) can also be obtained via the Legendre–Fenchel duality of Equation (1), implying that the optimization problem (4) corresponds to approximately maximizing the Legendre–Fenchel lower-bound [15].

2.2.2. Convergence Analysis

Here we briefly review theoretical convergence properties of LSMI.

First, let us consider the case where the function class $\mathcal{G}$ from which the function g is searched is a parametric model:

$$\mathcal{G} = \{g_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{y}) \mid \boldsymbol{\theta} \in \Theta \subset \mathbb{R}^b\}$$

Suppose that the true density-ratio r is contained in the model $\mathcal{G}$, *i.e.*, there exists $\boldsymbol{\theta}^*$ $(\in \Theta)$ such that: $r = g_{\boldsymbol{\theta}^*}$. Then, it was shown [28] that, under the standard regularity conditions for consistency [for example, see Section 3.2.1 of 36], it holds that:

$$\mathrm{LSMI}'(\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n) - \mathrm{SMI}(\boldsymbol{X}, \boldsymbol{Y}) = \mathcal{O}_p(n^{-1/2})$$

where $\mathcal{O}_p$ denotes the asymptotic order in probability. This shows that LSMI' retains the optimality in terms of the order of convergence in n, because $\mathcal{O}_p(n^{-1/2})$ is the optimal convergence rate in the parametric setup [37].

Next, we consider non-parametric cases where the function class $\mathcal{G}$ is a reproducing kernel Hilbert space [38] on $\mathcal{X} \times \mathcal{Y}$. Let us consider a non-parametric version of the LSMI optimization problem:

$$\widehat{g} := \operatorname*{argmin}_{g \in \mathcal{G}} \left[\frac{1}{2n^2} \sum_{i,j=1}^n g(\boldsymbol{x}_i, \boldsymbol{y}_j)^2 - \frac{1}{n} \sum_{i=1}^n g(\boldsymbol{x}_i, \boldsymbol{y}_i) + \frac{\lambda_n}{2} \|g\|_{\mathcal{G}}^2 \right]$$

where $\| \cdot \|_{\mathcal{G}}^2$ denotes the norm in the reproducing kernel Hilbert space $\mathcal{G}$. In the above optimization problem, a regularizer $\|g\|_{\mathcal{G}}^2$ is included to avoid overfitting and $\lambda_n \geq 0$ is the regularization parameter.

Suppose that the true density-ratio function r is contained in the function space $\mathcal{G}$ and is bounded from above. Then, it was shown [28] that, if $\lambda_n \to 0$ and $\lambda_n^{-1} = o(n^{2/(2+\gamma)})$ where γ $(0 < \gamma < 2)$ denotes a complexity measure of the function space $\mathcal{G}$ based on the *bracketing entropy* (The larger the value of γ is, the more complex the function space $\mathcal{G}$ is) [see p.83 of 36], it holds that:

$$\mathrm{LSMI}'(\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n) - \mathrm{SMI}(\boldsymbol{X}, \boldsymbol{Y}) = \mathcal{O}_p\big(\max(\lambda_n, n^{-1/2})\big) \tag{9}$$

The conditions $\lambda_n \to 0$ and $\lambda_n^{-1} = o(n^{2/(2+\gamma)})$ roughly mean that the regularization parameter λ_n should be sufficiently small but not too small. Equation (9) means that the convergence rate of the non-parametric version can also be $\mathcal{O}_p(n^{-1/2})$ for an appropriate choice of λ_n, but the non-parametric method requires a milder model assumption. According to [15], the above convergence rate is the minimax optimal rate under some setup. Thus, the convergence property of the above non-parametric method would also be optimal in the same sense.

2.3. Practical Implementation of LSMI

We have seen that LSMI has desirable convergence properties. Here we review practical implementation of LSMI. A MATLAB® implementation of LSMI is publicly available [39].

2.3.1. LSMI for Linear-in-Parameter Models

Let us approximate the density ratio Equation (2) using the following linear-in-parameter model:

$$g_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{y}) = \sum_{\ell=1}^{b} \theta_\ell \phi_\ell(\boldsymbol{x}, \boldsymbol{y}) = \boldsymbol{\theta}^\top \boldsymbol{\phi}(\boldsymbol{x}, \boldsymbol{y}) \tag{10}$$

where $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_b)^\top$ are parameters, $\boldsymbol{\phi}(\boldsymbol{x}, \boldsymbol{y}) = (\phi_1(\boldsymbol{x}, \boldsymbol{y}), \ldots, \phi_b(\boldsymbol{x}, \boldsymbol{y}))^\top$ are fixed basis functions, and $^\top$ denotes the transpose. Practical choices of the basis functions will be explained in Section 2.3.2. . For this model, the LSMI optimization problem with an ℓ_2-regularizer is expressed as:

$$\widehat{\boldsymbol{\theta}} := \mathop{\mathrm{argmin}}_{\boldsymbol{\theta} \in \mathbb{R}^b} \left[\frac{1}{2} \boldsymbol{\theta}^\top \widehat{\boldsymbol{H}} \boldsymbol{\theta} - \boldsymbol{\theta}^\top \widehat{\boldsymbol{h}} + \frac{\lambda}{2} \boldsymbol{\theta}^\top \boldsymbol{\theta} \right]$$

where $\lambda \geq 0$ is the regularization parameter that controls the strength of regularization, $\widehat{\boldsymbol{H}}$ is the $b \times b$ matrix defined by:

$$\widehat{\boldsymbol{H}} := \frac{1}{n^2} \sum_{i,j=1}^{n} \boldsymbol{\phi}(\boldsymbol{x}_i, \boldsymbol{y}_j) \boldsymbol{\phi}(\boldsymbol{x}_i, \boldsymbol{y}_j)^\top$$

and $\widehat{\boldsymbol{h}}$ is the b-dimensional vector defined by:

$$\widehat{\boldsymbol{h}} := \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{\phi}(\boldsymbol{x}_i, \boldsymbol{y}_i)$$

The solution $\widehat{\boldsymbol{\theta}}$ can be analytically obtained as:

$$\widehat{\boldsymbol{\theta}} = (\widehat{\boldsymbol{H}} + \lambda \boldsymbol{I}_b)^{-1}\widehat{\boldsymbol{h}} \tag{11}$$

where $\boldsymbol{I}_b$ is the b-dimensional identity matrix. Finally, LSMI is also given analytically as:

$$\mathrm{LSMI}(\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n) = \frac{1}{2}\widehat{\boldsymbol{h}}^\top \widehat{\boldsymbol{\theta}} - \frac{1}{2} \tag{12}$$

or

$$\mathrm{LSMI}'(\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n) = -\frac{1}{2}\widehat{\boldsymbol{\theta}}^\top \widehat{\boldsymbol{H}}\widehat{\boldsymbol{\theta}} + \widehat{\boldsymbol{h}}^\top \widehat{\boldsymbol{\theta}} - \frac{1}{2} \tag{13}$$

Some elements of $\widehat{\boldsymbol{\theta}}$ may take negative values in the above formulation, which can lead to negative density-ratio values and negative LSMI values. Such negative values may be rounded up to zero if necessary, although this does not happen for sufficiently large n. Another option is to explicitly impose the non-negativity constraint $\theta_1, \ldots, \theta_b \geq 0$ on the optimization problem. However, by this modification, the solution can no longer be obtained analytically, but only numerically using a quadratic program solver. (In this case, if the ℓ_2-regularizer is replaced with the ℓ_1-regularizer, the regularization path [40,41]—*i.e.*, solutions for all different regularization parameter values—can be computed efficiently without a quadratic program solver just by solving systems of linear equation [23].)

2.3.2. Design of Basis Functions

The practical accuracy of LSMI depends on the choice of basis functions in the model Equation (10). A typical choice is a non-parametric kernel model, *i.e.*, setting the number of basis function to $b = n$ and the ℓ-th basis function to $\phi_\ell(\boldsymbol{x}, \boldsymbol{y}) = K(\boldsymbol{x}, \boldsymbol{x}_\ell)L(\boldsymbol{y}, \boldsymbol{y}_\ell)$:

$$g_{\boldsymbol{\theta}}(\boldsymbol{x}, \boldsymbol{y}) = \sum_{\ell=1}^{n} \theta_\ell K(\boldsymbol{x}, \boldsymbol{x}_\ell)L(\boldsymbol{y}, \boldsymbol{y}_\ell) \tag{14}$$

where $K(\boldsymbol{x}, \boldsymbol{x}')$ and $L(\boldsymbol{y}, \boldsymbol{y}')$ are kernel functions for $\boldsymbol{x}$ and $\boldsymbol{y}$, respectively. If n is too large, b may be set to be smaller than n and choose a subset of data points $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n$ as kernel centers.

For real vector $\boldsymbol{x} \in \mathbb{R}^d$, we may practically use the Gaussian kernel for $K(\boldsymbol{x}, \boldsymbol{x}')$ after element-wise variance normalization:

$$K(\boldsymbol{x}, \boldsymbol{x}') = \exp\left(-\frac{\|\boldsymbol{x} - \boldsymbol{x}'\|^2}{2\sigma_{\mathrm{x}}^2}\right)$$

where $\sigma_{\mathrm{x}} > 0$ is the Gaussian width. When $\boldsymbol{x}$ is a non-vectorial structured object such as a string, a tree, and a graph, we may employ a kernel function defined for such structured data [42].

In the (multi-output) regression scenario where $\boldsymbol{y}$ is a real vector, the Gaussian kernel may also be used for $L(\boldsymbol{y}, \boldsymbol{y}')$ after element-wise variance normalization:

$$L(\boldsymbol{y}, \boldsymbol{y}') = \exp\left(-\frac{\|\boldsymbol{y} - \boldsymbol{y}'\|^2}{2\sigma_{\mathrm{y}}^2}\right)$$

where $\sigma_{\mathrm{y}} > 0$ is the Gaussian width. In the multi-class classification scenario where $y \in \{1, \ldots, c\}$ and c denotes the number of classes, we may use the delta kernel for $L(y, y')$:

$$L(y, y') = \begin{cases} 1 & \text{if } y = y' \\ 0 & \text{if } y \neq y' \end{cases}$$

Note that, in the classification case with the delta kernel, the LSMI solution can be computed efficiently in a class-wise manner [33]. In the multi-label classification scenario where $\boldsymbol{y} \in \{0, 1\}^c$ and c denotes the number of labels, we may use the normalized linear kernel function [43] for $\boldsymbol{y}$:

$$L(\boldsymbol{y}, \boldsymbol{y}') = \frac{(\boldsymbol{y} - \overline{\boldsymbol{y}})^\top (\boldsymbol{y}' - \overline{\boldsymbol{y}})}{\|\boldsymbol{y} - \overline{\boldsymbol{y}}\| \|\boldsymbol{y}' - \overline{\boldsymbol{y}}'\|}$$

where $\overline{\boldsymbol{y}} = \frac{1}{n} \sum_{i=1}^{n} \boldsymbol{y}_i$ is the sample mean.

2.3.3. Model Selection by Cross-Validation

Most of the above kernels include tuning parameters such as the Gaussian width, and the practical performance of LSMI depends on the choice of such kernel parameters and the regularization parameter λ. Model selection of LSMI is possible based on cross-validation with respect to the criterion J defined by Equation (3).

More specifically, the sample set $\mathcal{D} = \{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^{n}$ is divided into M disjoint subsets $\{\mathcal{D}_m\}_{m=1}^{M}$. Then the LSMI solution $\widehat{g}_m(\boldsymbol{x})$ is obtained using $\mathcal{D} \backslash \mathcal{D}_m$ (*i.e.*, all samples without $\mathcal{D}_m$), and its J-score for the hold-out samples $\mathcal{D}_m$ is computed as:

$$\widehat{J}_m^{\mathrm{CV}} := \frac{1}{2|\mathcal{D}_m|^2} \sum_{\boldsymbol{x}, \boldsymbol{y} \in \mathcal{D}_m} \widehat{g}_m(\boldsymbol{x}, \boldsymbol{y})^2 - \frac{1}{|\mathcal{D}_m|} \sum_{(\boldsymbol{x}, \boldsymbol{y}) \in \mathcal{D}_m} \widehat{g}_m(\boldsymbol{x}, \boldsymbol{y})$$

where $|\mathcal{D}_m|$ denotes the number of elements in the set $\mathcal{D}_m$. $\sum_{\boldsymbol{x}, \boldsymbol{y} \in \mathcal{D}_m}$ denotes the summation over all combinations of $\boldsymbol{x}$ and $\boldsymbol{y}$ in $\mathcal{D}_m$ (and thus $|\mathcal{D}_m|^2$ terms), while $\sum_{(\boldsymbol{x}, \boldsymbol{y}) \in \mathcal{D}_m}$ denotes the summation over all pairs $(\boldsymbol{x}, \boldsymbol{y})$ in $\mathcal{D}_m$ (and thus $|\mathcal{D}_m|$ terms). This procedure is repeated for $m = 1, \ldots, M$, and the average score,

$$\widehat{J}^{\mathrm{CV}} := \frac{1}{M} \sum_{m=1}^{M} \widehat{J}_m^{\mathrm{CV}}$$

is computed. Finally, the model (the kernel parameter and the regularization parameter in the current setup) that minimizes $\widehat{J}^{\mathrm{CV}}$ is chosen as the most suitable one.

3. SMI-Based Machine Learning

In this section, we show how the SMI estimator, LSMI, can be used for solving various machine learning tasks.

3.1. Independence Testing

First, we show how the SMI estimator can be used for independence testing.

3.1.1. Introduction

Identifying the statistical independence between random variables is one of the fundamental challenges in statistical data analysis. A traditional independence measure between random variables is the Pearson correlation coefficient, which can be used for detecting linear dependency. Recently, kernel-based independence measures have been studied to detect non-linear dependency. The Hilbert–Schmidt independence criterion (HSIC) [44] utilizes cross-covariance operators on universal reproducing kernel Hilbert spaces (RKHSs) [45], which is an infinite-dimensional generalization of covariance matrices. HSIC allows efficient detection of non-linear dependency by making use of the reproducing property of RKHSs [38]. However, HSIC has a weakness that its performance depends on the choice of RKHSs and there is no theoretically justified way to determine the RKHS properly thus far. In practice, using the Gaussian RKHS with width set to the median distance between samples is a popular heuristic [46], but this does not always work well.

To overcome the above limitations, an SMI-based independence test called *least-squares independence test* (LSIT) was proposed [26]. Below, we review LSIT.

3.1.2. Independence Testing with SMI

Let $\boldsymbol{x} \in \mathcal{X}$ be an input feature and $\boldsymbol{y} \in \mathcal{Y}$ be an output feature, which follow a joint probability distribution with density $p(\boldsymbol{x}, \boldsymbol{y})$. Suppose that we are given a set of independent and identically distributed (i.i.d.) paired samples $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n$. The objective of independence testing is to conclude whether $\boldsymbol{x}$ and $\boldsymbol{y}$ are statistically independent or not, based on the samples $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n$.

The SMI-based independence test, where the null hypothesis is that $\boldsymbol{x}$ and $\boldsymbol{y}$ are statistically independent, is based on the permutation test procedure [47]. More specifically, LSMI is first run using the original dataset $\mathcal{D} = \{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n$, and an SMI estimate, $\mathrm{LSMI}(\mathcal{D})$, is obtained. Next, $\{\boldsymbol{y}_i\}_{i=1}^n$ are randomly permuted and a shuffled dataset $\widetilde{\mathcal{D}} = \{(\boldsymbol{x}_i, \widetilde{\boldsymbol{y}}_i)\}_{i=1}^n$ is formed, where $\{\widetilde{\boldsymbol{y}}_i\}_{i=1}^n$ denote permuted samples. Then LSMI is run again using the shuffled dataset $\widetilde{\mathcal{D}}$, and an SMI estimate $\mathrm{LSMI}(\widetilde{\mathcal{D}})$ is obtained. Note that the random permutation eliminates the dependency between $\boldsymbol{x}$ and $\boldsymbol{y}$ (if it exists), and therefore $\mathrm{LSMI}(\widetilde{\mathcal{D}})$ would take a value close to zero. This random permutation procedure is repeated many times, and the distribution of $\mathrm{LSMI}(\widetilde{\mathcal{D}})$ under the null-hypothesis that $\boldsymbol{x}$ and $\boldsymbol{y}$ are statistically independent is constructed. Finally, the p-value is approximated by evaluating the relative ranking of $\mathrm{LSMI}(\mathcal{D})$ in the distribution of $\mathrm{LSMI}(\widetilde{\mathcal{D}})$.

This procedure is called the *least-squares independence test* (LSIT) [26]. A MATLAB® implementation of LSIT is publicly available [48].

3.2. Supervised Feature Selection

Next, we show how the SMI estimator can be used for supervised feature selection.

3.2.1. Introduction

The objective of supervised learning is to learn an input-output relation from input-output paired samples. However, when the dimensionality of input vectors is large, using all input elements could lead to a model interpretability problem. Feature selection is aimed at finding a subset of input elements that is useful for predicting output values [49].

Feature ranking is a simple implementation of feature selection that ranks each feature according to its relevance. In this feature ranking scenario, SMI between a single input variable and an output was shown to be useful [19]. However, feature ranking does not take feature interaction into account, and thus it is not useful when each single feature is not capable of predicting outputs, but multiple features are necessary for a valid prediction of outputs (e.g., an XOR problem). Two criteria, relevancy and redundancy, are often used to select multiple features simultaneously: A feature is said to be relevant if it can explain outputs, and features are said to be redundant if they are similar. Ideally, we want to find a subset of features that has high relevance and low redundancy.

Another important issue in feature selection is the computational cost: Naively selecting multiple features causes computational infeasibility because the number of possible feature combinations is exponential with respect to the number of input features. To cope with this problem, a computationally efficient method to handle multiple features called the least absolute shrinkage and selection operator (LASSO) [50] was proposed. In LASSO, a predictor consisting of a weighted sum of each feature is fitted to output values using the least-squares method, while the weight vector is confined in an ℓ_1-ball. The ℓ_1-ball restriction actually provides a notable property that the solution is sparsified, meaning that some of the weight parameters become exactly zero. Thus, LASSO automatically removes irrelevant features from its predictor, which can be achieved through convex optimization in a computationally efficient way [51,52].

However, LASSO can only handle linear predictors and its feature selection characteristic explicitly depends on the squared-loss function used in the least-squares method. To go beyond these limitations, an SMI-based feature selection method called ℓ_1-*LSMI* was proposed [27]. Below, we review ℓ_1-LSMI.

3.2.2. Feature Selection with SMI

The objective of feature selection is, from input feature vector $\boldsymbol{x} = (x^{(1)}, \ldots, x^{(d)})^\top \in \mathbb{R}^d$, to choose a subset of its elements that are useful for the prediction of output $\boldsymbol{y} \in \mathcal{Y}$. Suppose that we are given n i.i.d. paired samples $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n$ drawn from a joint distribution with density $p(\boldsymbol{x}, \boldsymbol{y})$. Let $w_1, \ldots, w_d$ be feature weights for $x^{(1)}, \ldots, x^{(d)}$, and we learn the weights as:

$$\max_{w_1,\ldots,w_d} \quad \mathrm{LSMI}\left(\left\{\left((w_1 x_i^{(1)}, \ldots, w_d x_i^{(d)})^\top, \boldsymbol{y}_i\right)\right\}_{i=1}^n\right)$$

$$\text{subject to} \quad \sum_{i=1}^d w_i \leq \eta \text{ and } w_1, \ldots, w_d \geq 0$$

where $\eta \geq 0$ is the regularization parameter that controls the number of features. Because the sign of feature weights is not relevant in feature selection, they are restricted to be non-negative. For non-negative weights, $\sum_{i=1}^d w_i$ is reduced to the ℓ_1-norm of the feature weight vector $(w_1, \ldots, w_d)^\top$. The features having zero weights are regarded as irrelevant in this formulation.

To compute the solution, a simple gradient ascent may be used, where the weight vector is projected onto the positive orthant of the ℓ_1-ball in each iteration to guarantee the feasibility. This can be performed by first projecting the weight vector onto the positive orthant by rounding up negative elements to zero, and then projecting it onto the ℓ_1-ball [54].

This SMI-based feature selection algorithm is called the ℓ_1-*LSMI* [27]. A MATLAB® implementation of ℓ_1-LSMI is publicly available [53].

3.3. Supervised Feature Extraction

While feature selection chooses a subset of features for enhancing model interpretability, feature extraction finds a low-dimensional representation of features that can depend on all features (e.g., through linear combination) for improving the prediction accuracy. Here, we show how the SMI estimator can be used for supervised feature extraction.

3.3.1. Introduction

The goal of sufficient dimension reduction (SDR) is to map input features to low-dimensional expressions while "sufficient" information for predicting output values is maintained [55]. Earlier SDR methods developed in the statistics community, such as sliced inverse regression [56], principal Hessian direction [57], and sliced average variance estimation [58], rely on the ellipticity of the data (e.g., Gaussian), but such an assumption may not be fulfilled in practice. To overcome the limitations of these approaches, kernel dimension reduction (KDR) was proposed [59]. KDR employs a kernel-based dependence measure that is distribution-free, and thus KDR is flexible. However, it lacks systematic model selection strategies for kernel and regularization parameters. Furthermore, KDR scales poorly to massive datasets and there is no good way to set an initial solution for its gradient-based optimization. In practice, many restarts from different initial solutions may be needed for finding a good local optimum, which makes the entire procedure even slower and the performance of dimension reduction unreliable.

To overcome the above limitations, an SMI-based SDR method called *least-squares dimension reduction* (LSDR) was proposed [28]. An advantage of LSDR is that its tuning parameters can be systematically optimized based on cross-validation. To further address the computational and initialization issues, a heuristic search strategy for LSDR called *sufficient component analysis* (SCA) was proposed [29]. Below, we review LSDR and SCA.

3.3.2. Sufficient Dimension Reduction with SMI

First, we formulate the problem of SDR [55]. Let $\boldsymbol{x} \in \mathbb{R}^{d_\mathrm{x}}$ be an input vector and $\boldsymbol{y} \in \mathcal{Y}$ be an output. The goal of SDR is to find a subspace of input domain $\mathbb{R}^{d_\mathrm{x}}$ that contains "sufficient" information about output $\boldsymbol{y}$. We assume that there exists an orthogonal matrix $\boldsymbol{U}^* \in \mathbb{R}^{d_\mathrm{u} \times d_\mathrm{x}}$ for $d_\mathrm{u} \leq d_\mathrm{x}$ such that

$$\boldsymbol{y} \perp\!\!\!\perp \boldsymbol{x} \mid \boldsymbol{U}^* \boldsymbol{x} \tag{15}$$

That is, given the projected feature $\boldsymbol{U}^*\boldsymbol{x}$, the (remaining) feature $\boldsymbol{x}$ is conditionally independent of output $\boldsymbol{y}$ and thus can be discarded without sacrificing the predictability of $\boldsymbol{y}$. The objective of SDR is to find such $\boldsymbol{U}^*$ from n i.i.d. paired samples, $\{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n$, drawn from a joint distribution with density $p(\boldsymbol{x}, \boldsymbol{y})$. We assume that the projection dimensionality d_{u} is known.

SMI can be used for characterizing the optimal projection matrix $\boldsymbol{U}^*$ [28]. Indeed, it was shown that inequality,

$$\mathrm{SMI}(\boldsymbol{X}, \boldsymbol{Y}) \geq \mathrm{SMI}(\boldsymbol{U}\boldsymbol{X}, \boldsymbol{Y})$$

holds, and the equality holds if and only if Equation (15) holds. Thus, maximizing $\mathrm{SMI}(\boldsymbol{U}\boldsymbol{X}, \boldsymbol{Y})$ with respect to $\boldsymbol{U}$ leads to $\boldsymbol{U}^*$. In practice, the following optimization problem is solved:

$$\max_{\boldsymbol{U} \in \mathbb{R}^{d_{\mathrm{u}} \times d_{\mathrm{x}}}} \mathrm{LSMI}(\{(\boldsymbol{U}\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n)$$
$$\text{subject to } \boldsymbol{U}\boldsymbol{U}^\top = \boldsymbol{I}_{d_{\mathrm{u}}}$$

This formulation is called *least-squares dimension reduction* (LSDR) [28].

3.3.3. Gradient-Based Subspace Search

A simple approach to solving the above LSDR optimization problem is the following iterative procedure:

- $\boldsymbol{U}$ is updated to ascend the gradient of $\mathrm{LSMI}(\{(\boldsymbol{U}\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n)$ with respect to $\boldsymbol{U}$.
- $\boldsymbol{U}$ is projected onto the feasible region specified by $\boldsymbol{U}\boldsymbol{U}^\top = \boldsymbol{I}_{d_{\mathrm{u}}}$.

The gradient of $\mathrm{LSMI}(\{(\boldsymbol{U}\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n)$ with respect to $\boldsymbol{U}$ is given by:

$$\frac{\partial \mathrm{LSMI}}{\partial \boldsymbol{U}} = \sum_{\ell=1}^{b} \widehat{\theta}_\ell \frac{\partial \widehat{h}_\ell}{\partial \boldsymbol{U}} - \frac{1}{2} \sum_{\ell,\ell'=1}^{b} \widehat{\theta}_\ell \widehat{\theta}_{\ell'} \frac{\partial \widehat{H}_{\ell,\ell'}}{\partial \boldsymbol{U}}$$

If the kernel model Equation (14) with the Gaussian kernel,

$$K(\boldsymbol{U}\boldsymbol{x}, \boldsymbol{U}\boldsymbol{x}') = \exp\left(-\frac{\|\boldsymbol{U}\boldsymbol{x} - \boldsymbol{U}\boldsymbol{x}'\|^2}{2\sigma^2}\right)$$

is used, $\frac{\partial \widehat{h}_\ell}{\partial \boldsymbol{U}}$ and $\frac{\partial \widehat{H}_{\ell,\ell'}}{\partial \boldsymbol{U}}$ (for $\ell, \ell' = 1, \ldots, n$) are given by:

$$\frac{\partial \widehat{h}_\ell}{\partial \boldsymbol{U}} = -\frac{1}{n\sigma^2} \sum_{i=1}^{n} (\boldsymbol{U}\boldsymbol{x}_i - \boldsymbol{U}\boldsymbol{x}_\ell)(\boldsymbol{x}_i - \boldsymbol{x}_\ell)^\top \exp\left(-\frac{\|\boldsymbol{U}\boldsymbol{x}_i - \boldsymbol{U}\boldsymbol{x}_\ell\|^2}{2\sigma^2}\right) L(\boldsymbol{y}_i, \boldsymbol{y}_\ell),$$
$$\frac{\partial \widehat{H}_{\ell,\ell'}}{\partial \boldsymbol{U}} = \left[-\frac{1}{n\sigma^2} \sum_{i=1}^{n} \left((\boldsymbol{U}\boldsymbol{x}_i - \boldsymbol{U}\boldsymbol{x}_\ell)(\boldsymbol{x}_i - \boldsymbol{x}_\ell)^\top + (\boldsymbol{U}\boldsymbol{x}_i - \boldsymbol{U}\boldsymbol{x}_{\ell'})(\boldsymbol{x}_i - \boldsymbol{x}_{\ell'})^\top \right) \right.$$
$$\left. \times \exp\left(-\frac{\|\boldsymbol{U}\boldsymbol{x}_i - \boldsymbol{U}\boldsymbol{x}_\ell\|^2 + \|\boldsymbol{U}\boldsymbol{x}_i - \boldsymbol{U}\boldsymbol{x}_{\ell'}\|^2}{2\sigma^2}\right) \right] \times \left[\frac{1}{n} \sum_{i=1}^{n} L(\boldsymbol{y}_i, \boldsymbol{y}_\ell) L(\boldsymbol{y}_i, \boldsymbol{y}_{\ell'}) \right]$$

The projection of $\boldsymbol{U}$ onto the feasible region specified by $\boldsymbol{U}\boldsymbol{U}^\top = \boldsymbol{I}_{d_{\mathrm{u}}}$ may be carried out by the Gram–Schmidt process [60], although this is time-consuming.

An alternative way to solve the LSDR optimization problem is to perform gradient ascent on the Grassmann manifold [61]. In the Euclidean space, the ordinary gradient gives the steepest direction.

However, on a manifold, the natural gradient [62] gives the steepest direction. The natural gradient $\nabla\mathrm{LSMI}(\boldsymbol{U})$ at $\boldsymbol{U}$ is given as follows [63]:

$$\nabla\mathrm{LSMI}(\boldsymbol{U}) = \frac{\partial\mathrm{LSMI}}{\partial\boldsymbol{U}} - \frac{\partial\mathrm{LSMI}}{\partial\boldsymbol{U}}\boldsymbol{U}^\top\boldsymbol{U} = \frac{\partial\mathrm{LSMI}}{\partial\boldsymbol{U}}\boldsymbol{U}_\perp^\top\boldsymbol{U}_\perp$$

where $\boldsymbol{U}_\perp$ is any $(d-m)\times d$ matrix such that $[\boldsymbol{U}^\top\ \boldsymbol{U}_\perp^\top]$ is orthogonal. Then the geodesic from $\boldsymbol{U}$ to the direction of the natural gradient $\nabla\mathrm{LSMI}(\boldsymbol{U})$ over the Grassmann manifold can be expressed using $t \in \mathbb{R}$ as:

$$\boldsymbol{U}_t := \begin{bmatrix}\boldsymbol{I}_{d_{\mathrm{x}}} & \boldsymbol{O}_{d_{\mathrm{x}}-d_{\mathrm{u}}}\end{bmatrix} \exp\left(t \begin{bmatrix} \boldsymbol{O}_{d_{\mathrm{u}}} & \frac{\partial\mathrm{LSMI}}{\partial\boldsymbol{U}}\boldsymbol{U}_\perp^\top \\ -\boldsymbol{U}_\perp\frac{\partial\mathrm{LSMI}}{\partial\boldsymbol{U}}^\top & \boldsymbol{O}_{d_{\mathrm{x}}-d_{\mathrm{u}}} \end{bmatrix}\right) \begin{bmatrix}\boldsymbol{U} \\ \boldsymbol{U}_\perp\end{bmatrix}$$

where "exp" for a matrix denotes the matrix exponential, and $\boldsymbol{O}_{d_{\mathrm{x}}}$ is the $d_{\mathrm{x}} \times d_{\mathrm{x}}$ zero matrix. Thus, line search along the geodesic in the natural gradient direction is equivalent to finding the maximizer from $\{\boldsymbol{U}_t \mid t \geq 0\}$. For choosing the step size of each gradient update, some approximate line search method such as Armijo's rule [64] or backtracking line search [51] may be used.

A MATLAB® implementation of LSDR is publicly available [65].

3.3.4. Heuristic Subspace Search

Although the natural gradient method is computationally more efficient than the plain gradient method, it is still time consuming. Moreover, many restarts from different initial solutions may be needed for finding a good local optimum. Here, we introduce a heuristic method that can address these issues [29].

A key idea is to use a truncated negative quadratic function called the Epanechnikov kernel [66] as a kernel function for $\boldsymbol{U}\boldsymbol{x}$:

$$K(\boldsymbol{U}\boldsymbol{x}, \boldsymbol{U}\boldsymbol{x}') = \max\left(0, 1 - \frac{\|\boldsymbol{U}\boldsymbol{x} - \boldsymbol{U}\boldsymbol{x}'\|^2}{2\sigma_{\mathrm{z}}^2}\right)$$

Let $I(c)$ be the indicator function, *i.e.*, $I(c) = 1$ if c is true and zero otherwise. Then, for the above kernel function, LSMI can be expressed as:

$$\mathrm{LSMI} = \frac{1}{2}\mathrm{tr}(\boldsymbol{U}\boldsymbol{D}_{\boldsymbol{U}}\boldsymbol{U}^\top) - \frac{1}{2}$$

where $\mathrm{tr}(\cdot)$ denotes the trace of a matrix and $\boldsymbol{D}_{\boldsymbol{U}}$ is the $d_{\mathrm{x}} \times d_{\mathrm{x}}$ matrix defined by:

$$\boldsymbol{D}_{\boldsymbol{U}} = \frac{1}{n}\sum_{i=1}^{n}\sum_{\ell=1}^{n}\widehat{\theta}_\ell(\boldsymbol{U}) I\left(\frac{\|\boldsymbol{U}\boldsymbol{x}_i - \boldsymbol{U}\boldsymbol{x}_\ell\|^2}{2\sigma_{\mathrm{z}}^2} < 1\right) L(\boldsymbol{y}_i, \boldsymbol{y}_\ell)\left[\frac{1}{d_{\mathrm{u}}}\boldsymbol{I}_{d_{\mathrm{x}}} - \frac{1}{2\sigma_{\mathrm{z}}^2}(\boldsymbol{x}_i - \boldsymbol{x}_\ell)(\boldsymbol{x}_i - \boldsymbol{x}_\ell)^\top\right]$$

Here, the fact that $\widehat{\theta}_\ell$ depends on $\boldsymbol{U}$ is explicitly indicated by $\widehat{\theta}_\ell(\boldsymbol{U})$.

If $\boldsymbol{U}$ in $\boldsymbol{D}_{\boldsymbol{U}}$ is replaced by $\boldsymbol{U}'$, where $\boldsymbol{U}'$ is a transformation matrix obtained in the previous iteration, the SMI estimator is simplified as:

$$\frac{1}{2}\mathrm{tr}\left(\boldsymbol{U}\boldsymbol{D}_{\boldsymbol{U}'}\boldsymbol{U}^\top\right) - \frac{1}{2} \tag{16}$$

Because $\boldsymbol{D}_{\boldsymbol{U}'}$ is independent of $\boldsymbol{U}$, a maximizer of Equation (16) with respect to $\boldsymbol{U}$ can be analytically obtained by $(\boldsymbol{u}_1|\cdots|\boldsymbol{u}_{d_{\mathrm{u}}})^\top$, where $\{\boldsymbol{u}_i\}_{i=1}^{d_{\mathrm{u}}}$ are the d_{u} principal components of $\boldsymbol{D}'$. The same technique can also be utilized for determining an initial transformation matrix, by computing the above solution for $\boldsymbol{U}' = \boldsymbol{I}_{d_{\mathrm{x}}}$ (*i.e.*, no dimensionality reduction).

The above heuristic search method for LSDR is called *sufficient component analysis* (SCA) [29]. A MATLAB® implementation of SCA is publicly available [67].

3.4. Canonical Dependency Analysis

Next, we show how the SMI estimator can be used for feature extraction from two sets of data.

3.4.1. Introduction

Canonical correlation analysis (CCA) [68] is a classical dimensionality reduction technique for two data sources, and it iteratively finds projection directions with maximum correlation. However, because CCA only captures correlations under linear projections, it is often insufficient to analyze complex real-world data that contain higher-order correlations. To be more flexible, non-linear CCA methods have been explored. A simple approach uses neural networks to handle non-linear projections [69,70], but neural networks are prone to local optima. Another approach first non-linearly transforms data samples into feature spaces and then apply linear CCA [71,72]. Given that the non-linear transformation is fixed, this two-step approach allows analytic computation of the global optimal solution via a generalized eigenvalue problem in the same way as linear CCA. This non-linear approach is called kernel CCA (KCCA) because reproducing kernels [38] are used as non-linear transforms. Alternating regression such as the alternating conditional expectation [73] is another possible way to find dependency in a flexible manner, which estimates transformations for two variables alternately by minimizing the squared error between transformed variables. These non-linear variants of CCA are highly flexible, although obtained results are often difficult to interpret due to the non-linearity.

The above non-linear CCA approaches can be regarded as capturing correlations along non-linear projection directions. Another extension of CCA called canonical dependency analysis (CDA) [30] captures higher-order correlations under linear projections. It was shown that KCCA with a universal kernel [45] such as the Gaussian kernel allows efficient detection of higher-order correlations [74]. However, the choice of universal kernels affects the practical performance, and there is no systematic method to choose a suitable kernel function. Another approach to higher-order CCA called informational CCA (ICCA) [75] uses mutual information (MI) as a dependency measure, where MI is estimated via kernel density estimation (KDE). Because systematic model selection strategies are available for KDE [76], ICCA could be more practical than the KCCA-based CDA method. In the ICCA method, one-dimensional projection directions are found in an iterative manner. Thus, it would be more powerful if multi-dimensional projection directions (*i.e.*, a subspace) could be directly found in CDA [30]. However, ICCA may not be reliable in such a subspace search scenario because it involves the ratio of estimated densities, which tends to produce large estimation error if the dimensionality is not small.

To overcome the above limitation, an SMI-based CDA method called *least-squares CDA* (LSCDA) was proposed [30]. Below, we review LSCDA.

3.4.2. Canonical Dependency Analysis with SMI

Suppose that we are given n i.i.d. paired samples $\{(\boldsymbol{x}_i, \boldsymbol{y}_i) \mid \boldsymbol{x}_i \in \mathbb{R}^{d_{\mathrm{x}}},\ \boldsymbol{y}_i \in \mathbb{R}^{d_{\mathrm{y}}}\}_{i=1}^{n}$ drawn from a joint distribution with density $p(\boldsymbol{x}, \boldsymbol{y})$. CDA is aimed at finding the low-dimensional expressions of $\boldsymbol{x}$ and $\boldsymbol{y}$ that are maximally dependent on each other. Here, we focus on linear dimension reduction, *i.e.*, $\boldsymbol{x}$ and $\boldsymbol{y}$ are transformed as $\boldsymbol{U}\boldsymbol{x}$ and $\boldsymbol{V}\boldsymbol{y}$, where $\boldsymbol{U} \in \mathbb{R}^{d_{\mathrm{u}} \times d_{\mathrm{x}}}$ and $\boldsymbol{V} \in \mathbb{R}^{d_{\mathrm{v}} \times d_{\mathrm{y}}}$ are orthogonal matrices with known dimensionalities d_{u} and d_{v}. The objective of CDA is to find the transformation matrices $\boldsymbol{U}$ and $\boldsymbol{V}$ such that the statistical dependency between $\boldsymbol{U}\boldsymbol{x}$ and $\boldsymbol{V}\boldsymbol{y}$ is maximized. Let us use the SMI estimator, $\mathrm{LSMI}(\{(\boldsymbol{U}\boldsymbol{x}_i, \boldsymbol{V}\boldsymbol{y}_i)\}_{i=1}^{n})$, as the dependency measure, *i.e.*, we solve,

$$\begin{aligned} &\mathop{\mathrm{argmax}}_{\boldsymbol{U} \in \mathbb{R}^{d_{\mathrm{u}} \times d_{\mathrm{x}}}, \boldsymbol{V} \in \mathbb{R}^{d_{\mathrm{v}} \times d_{\mathrm{y}}}} && \mathrm{LSMI}(\{(\boldsymbol{U}\boldsymbol{x}_i, \boldsymbol{V}\boldsymbol{y}_i)\}_{i=1}^{n}) \\ &\text{subject to} && \boldsymbol{U}\boldsymbol{U}^{\top} = \boldsymbol{I}_{d_{\mathrm{u}}} \text{ and } \boldsymbol{V}\boldsymbol{V}^{\top} = \boldsymbol{I}_{d_{\mathrm{v}}} \end{aligned}$$

This formulation is called *least-squares CDA* (LSCDA) [30].

The above optimization problem can be solved in the same way as LSDR presented in Section 3.3.3. A MATLAB® implementation of LSCDA is publicly available [77].

3.5. Independent Component Analysis

Here, we show how the SMI estimator can be used for independent component analysis.

3.5.1. Introduction

Suppose that there exist statistically independent sources of signals, and we observe their mixtures. The purpose of independent component analysis (ICA) [78] is to separate the mixed signals into the original source signals. An approach to ICA is to separate the mixed signals such that statistical independence among separated signals is maximized under some independence measure.

Various methods for evaluating the statistical independence among random variables from samples have been explored so far. A naive approach is to estimate probability densities based on parametric or non-parametric density estimation methods. However, finding an appropriate parametric model is not straightforward without strong prior knowledge and non-parametric estimation is not generally accurate in high-dimensional problems. Thus, this naive approach is not reliable in practice. Another approach is to approximate the entropy based on the Gram–Charlier expansion [79] or the Edgeworth expansion [80]. An advantage of this entropy-based approach is that a hard task of density estimation is not directly involved. However, these expansion techniques are based on the assumption that the target density is close to Gaussian, and violation of this assumption can cause large approximation error.

The above approaches are based on the probability densities of signals. Another line of research that does not explicitly involve probability densities employs non-linear correlation—signals are statistically independent if and only if all non-linear correlations among signals vanish. Following this line, computationally efficient algorithms have been developed based on a contrast function [81,82], which is an approximation of the entropy or mutual information. However, non-linearities in the contrast function need to be pre-specified in these methods, and thus they could be inaccurate if the predetermined non-linearities do not match the target distribution. To cope with this problem, the kernel trick has been applied in ICA, which allows computationally efficient evaluation of all non-linear correlations citeJMLR:Bach+Jordan:2002. However, its practical performance depends on the choice of kernels (more specifically, the Gaussian kernel width) and there seems no theoretically justified method to determine the kernel width. This is a critical problem in unsupervised learning tasks such as ICA.

To cope with this problem, an SMI-based ICA algorithm called *least-squares independent component analysis* (LICA) has been developed [31]. Below, we review LICA.

3.5.2. Independent Component Analysis with SMI

Suppose there are d signal sources and let: $\{\boldsymbol{x}_i \mid \boldsymbol{x}_i = (x_i^{(1)}, \ldots, x_i^{(d)})^\top \in \mathbb{R}^d\}_{i=1}^n$ be i.i.d. samples drawn from a distribution with density $p(\boldsymbol{x})$. We assume that elements $x^{(1)}, \ldots, x^{(d)}$ are statistically independent of each other, *i.e.*, $p(\boldsymbol{x})$ is factorized as:

$$p(\boldsymbol{x}) = p(x^{(1)}) \cdots p(x^{(d)})$$

We cannot directly observe $\{\boldsymbol{x}_i\}_{i=1}^n$, but only their linearly mixed samples $\{\boldsymbol{y}_i\}_{i=1}^n$:

$$\boldsymbol{y}_i := \boldsymbol{U}\boldsymbol{x}_i$$

where $\boldsymbol{U}$ is a $d \times d$ invertible matrix called the mixing matrix.

The goal of ICA is, from the mixed samples $\{\boldsymbol{y}_i\}_{i=1}^n$, to obtain a demixing matrix $\boldsymbol{V}$ that recovers the original source samples $\{\boldsymbol{x}_i\}_{i=1}^n$. We denote the demixed samples by $\{\boldsymbol{z}_i\}_{i=1}^n$:

$$\boldsymbol{z}_i = \boldsymbol{V}\boldsymbol{y}_i$$

The ideal solution is $\boldsymbol{V} = \boldsymbol{U}^{-1}$, but we can only recover the source signals up to permutation and scaling of components of $\boldsymbol{x}$ due to non-identifiability of the ICA setup [78]. Let us denote the demixed samples by:

$$\boldsymbol{z}_i = (z_i^{(1)}, \ldots, z_i^{(d)})^\top := \boldsymbol{V}\boldsymbol{y}_i$$

for $i = 1, \ldots, n$.

A direct approach to ICA is to determine $\boldsymbol{V}$ so that elements of $\boldsymbol{z}$ are as statistically independent as possible. Here, we adopt SMI as the independence measure:

$$\mathrm{SMI}(Z^{(1)}, \ldots, Z^{(d)}) := \frac{1}{2} \int \cdots \int p(z^{(1)}) \cdots p(z^{(d)}) \left(\frac{p(z^{(1)}, \ldots, z^{(d)})}{p(z^{(1)}) \cdots p(z^{(d)})} - 1 \right)^2 \mathrm{d}z^{(1)} \cdots \mathrm{d}z^{(d)}$$

We try to find the demixing matrix $\boldsymbol{V}$ that minimizes SMI. In practice, the following optimization problem is solved:

$$\min_{\boldsymbol{V} \in \mathbb{R}^{d \times d}} \ \mathrm{LSMI}(\{\boldsymbol{V}\boldsymbol{y}_i\}_{i=1}^n)$$

where $\mathrm{LSMI}(\{\boldsymbol{V}\boldsymbol{y}_i\}_{i=1}^n)$ is given by the same form as Equation (12) (or Equation (13)), but the matrix $\widehat{\boldsymbol{H}}$ and the vector $\widehat{\boldsymbol{h}}$ are defined in a slightly different way. For the Gaussian kernel,

$$K(\boldsymbol{V}\boldsymbol{y}, \boldsymbol{V}\boldsymbol{y}') = \exp\left(-\frac{\|\boldsymbol{V}\boldsymbol{y} - \boldsymbol{V}\boldsymbol{y}'\|^2}{2\sigma^2} \right)$$

$\widehat{\boldsymbol{H}}$ and $\widehat{\boldsymbol{h}}$ are given by:

$$\widehat{H}_{\ell,\ell'} = \frac{1}{n^d} \prod_{m=1}^{d} \left[\sum_{i=1}^{n} \exp\left(-\frac{(z_\ell^{(m)} - z_i^{(m)})^2 + (z_{\ell'}^{(m)} - z_i^{(m)})^2}{2\sigma^2} \right) \right]$$

$$\widehat{h}_\ell = \frac{1}{n} \sum_{i=1}^{n} \exp\left(-\frac{\|\boldsymbol{z}_i - \boldsymbol{z}_\ell\|^2}{2\sigma^2} \right)$$

This formulation is called *least-squares independent component analysis* (LICA) [31].

3.5.3. Gradient-Based Demixing Matrix Search

Based on the plain gradient technique, an update rule of $\boldsymbol{V}$ is given by:

$$\boldsymbol{V} \longleftarrow \boldsymbol{V} - t \frac{\partial \mathrm{LSMI}}{\partial \boldsymbol{V}} \tag{17}$$

where $t \ (> 0)$ is the step size. The gradient $\frac{\partial \mathrm{LSMI}}{\partial \boldsymbol{V}}$ is given by:

$$\frac{\partial \mathrm{LSMI}}{\partial \boldsymbol{V}} = \sum_{\ell=1}^{n} \widehat{\theta}_\ell \frac{\partial \widehat{h}_\ell}{\partial \boldsymbol{V}} - \frac{1}{2} \sum_{\ell,\ell'=1}^{n} \widehat{\theta}_\ell \widehat{\theta}_{\ell'} \frac{\partial \widehat{H}_{\ell,\ell'}}{\partial \boldsymbol{V}}$$

where

$$\frac{\partial \widehat{h}_\ell}{\partial V_{k,k'}} = -\frac{1}{n\sigma^2} \sum_{i=1}^{n} (z_i^{(k)} - z_\ell^{(k)})(y_i^{(k')} - y_\ell^{(k')})^\top \exp\left(-\frac{\|\boldsymbol{z}_i - \boldsymbol{z}_k\|^2}{2\sigma^2} \right)$$

$$\begin{aligned}\frac{\partial \widehat{H}_{\ell,\ell'}}{\partial V_{k,k'}} = {} & \frac{1}{n^{d-1}} \prod_{m \neq k} \left[\sum_{i=1}^{n} \exp\left(-\frac{(z_i^{(m)} - z_\ell^{(m)})^2 + (z_i^{(m)} - z_{\ell'}^{(m)})^2}{2\sigma^2} \right) \right] \\ & \times \left[-\frac{1}{n\sigma^2} \sum_{i=1}^{n} \left((z_i^{(k)} - z_\ell^{(k)})(y_i^{(k')} - y_\ell^{(k')}) + (z_i^{(k)} - z_{\ell'}^{(k)})(y_i^{(k')} - y_{\ell'}^{(k')}) \right) \right. \\ & \left. \times \exp\left(-\frac{(z_i^{(k)} - v_\ell^{(k)})^2 + (z_i^{(k)} - z_{\ell'}^{(k)})^2}{2\sigma^2} \right) \right]\end{aligned}$$

In ICA, scaling of components of $\boldsymbol{z}$ can be arbitrary. This implies that the above gradient updating rule can lead to a solution with poor scaling, which is not preferable from a numerical viewpoint. To avoid possible numerical instability, $\boldsymbol{V}$ is normalized at each gradient iteration as:

$$V_{k,k'} \longleftarrow \frac{V_{k,k'}}{\sqrt{\sum_{m=1}^{d} V_{k,m}^2}}$$

3.5.4. Natural Gradient Demixing Matrix Search

Suppose that data samples are whitened, *i.e.*, samples $\{\boldsymbol{y}_i\}_{i=1}^n$ are pre-transformed as:

$$\boldsymbol{y}_i \longleftarrow \widehat{\boldsymbol{\Sigma}}^{-\frac{1}{2}} \boldsymbol{y}_i$$

where $\widehat{\boldsymbol{\Sigma}}$ is the sample covariance matrix:

$$\widehat{\boldsymbol{\Sigma}} := \frac{1}{n} \sum_{i=1}^{n} \left(\boldsymbol{y}_i - \frac{1}{n} \sum_{j=1}^{n} \boldsymbol{y}_j \right) \left(\boldsymbol{y}_i - \frac{1}{n} \sum_{j=1}^{n} \boldsymbol{y}_j \right)^\top$$

Then it can be shown that any demixing matrix that eliminates the second order correlation is an orthogonal matrix [78]. Thus, for whitened data, the search space of $\boldsymbol{V}$ can be restricted to the orthogonal group without loss of generality. The natural gradient [62] update rule on the orthogonal group is given by:

$$\boldsymbol{V} \longleftarrow \boldsymbol{V} \exp\left(-t\left(\boldsymbol{V}^\top \frac{\partial \mathrm{LSMI}}{\partial \boldsymbol{V}} - \frac{\partial \mathrm{LSMI}}{\partial \boldsymbol{V}}^\top \boldsymbol{V}\right)\right)$$

where "exp" for a matrix denotes the matrix exponential and $t\ (> 0)$ is the step size.

A MATLAB® implementation of LICA is publicly available [83].

3.6. Cross-Domain Object Matching

Next, we show how the SMI estimator can be used for cross-domain object matching.

3.6.1. Introduction

The objective of cross-domain object matching is to match two sets of unpaired objects in different domains. For example, in photo album summarization, we are given a set of photos and a designed photo frame expressed as a set of photo slots in the Cartesian coordinate system, and we want to automatically assign the photos into the designed photo frame. A typical approach of cross-domain object matching is to find a mapping from objects in one domain (photos) to objects in the other domain (frame) so that the pairwise dependency is maximized. In this scenario, accurately evaluating the dependence between objects is a key issue.

Kernelized sorting [84] tries to find the mapping between two domains that maximizes mutual information under the Gaussian assumption. However, because the Gaussian assumption may not be fulfilled in practice, this method tends to perform poorly. To overcome the above limitation, the kernel-based dependence measure called the Hilbert–Schmidt independence criterion (HSIC) [85] was proposed to use in kernelized sorting [86]. Because HSIC is distribution-free, HSIC-based kernelized sorting is more flexible than the original method based on the Gaussian assumption. However, HSIC includes a tuning parameter (more specifically, the Gaussian kernel width), and its choice is crucial to obtain better performance [87].

To cope with this problem, an SMI-based cross-domain object matching method called *least-squares object matching* (LSOM) was developed [32]. Below, we review LSOM.

3.6.2. Cross-Domain Object Matching with SMI

The goal of cross-domain object matching is, given two sets of *unpaired* samples of the same size, $\{\boldsymbol{x}_i \mid \boldsymbol{x}_i \in \mathcal{X}\}_{i=1}^n$ and $\{\boldsymbol{y}_i \mid \boldsymbol{y}_i \in \mathcal{Y}\}_{i=1}^n$, to find a mapping that well "matches" them. Let π be a permutation function over $\{1, \ldots, n\}$. The optimal permutation, denoted by π^*, can be obtained as the maximizer of the dependency between the two sets $\{\boldsymbol{x}_i\}_{i=1}^n$ and $\{\boldsymbol{y}_{\pi(i)}\}_{i=1}^n$. Here, we use the SMI approximator, $\mathrm{LSMI}(\{(\boldsymbol{x}_i, \boldsymbol{y}_{\pi(i)})\}_{i=1}^n)$, as the dependency measure, *i.e.*, we solve,

$$\max_{\pi} \mathrm{LSMI}(\{(\boldsymbol{x}_i, \boldsymbol{y}_{\pi(i)})\}_{i=1}^n)$$

Let $\boldsymbol{K}$ and $\boldsymbol{L}$ be the $n \times n$ kernel matrices defined by $K_{i,j} = K(\boldsymbol{x}_i, \boldsymbol{x}_j)$ and $L_{i,j} = L(\boldsymbol{y}_i, \boldsymbol{y}_j)$. Then LSMI for $\{(\boldsymbol{x}_i, \boldsymbol{y}_{\pi(i)})\}_{i=1}^n$ can be expressed as:

$$\mathrm{LSMI}(\{(\boldsymbol{x}_i, \boldsymbol{y}_{\pi(i)})\}_{i=1}^n) = \frac{1}{2n}\mathrm{tr}\left(\boldsymbol{\Pi}^\top \boldsymbol{L}\boldsymbol{\Pi}\widehat{\boldsymbol{\Theta}}_{\boldsymbol{\Pi}}\boldsymbol{K}\right) - \frac{1}{2} \tag{18}$$

where $\boldsymbol{\Pi}$ is the permutation matrix corresponding to π, *i.e.*, $\boldsymbol{\Pi}$ is the $n \times n$ zero-one matrix such that $\Pi_{i,j} = 1$ if $i = \pi(j)$ for $j = 1, \ldots, n$ and $\Pi_{i,j} = 0$ otherwise. $\widehat{\boldsymbol{\Theta}}_{\boldsymbol{\Pi}}$ is the diagonal matrix with diagonal elements given by the LSMI solution $\widehat{\boldsymbol{\theta}}_\pi$ obtained by paired data $\{(\boldsymbol{x}_i, \boldsymbol{y}_{\pi(i)})\}_{i=1}^n$ (see Equation (11)).

Because maximizing Equation (18) with respect to $\boldsymbol{\Pi}$ is computationally infeasible, greedy update from previous solution $\boldsymbol{\Pi}'$ is used in practice:

$$\boldsymbol{\Pi}^{\mathrm{new}} = (1-t)\boldsymbol{\Pi}' + t \cdot \underset{\boldsymbol{\Pi}}{\mathrm{argmax}}\ \mathrm{tr}\left(\boldsymbol{\Pi}^\top \boldsymbol{L}\boldsymbol{\Pi}'\widehat{\boldsymbol{\Theta}}_{\boldsymbol{\Pi}'}\boldsymbol{K}\right)$$

where $0 < t \leq 1$ is the step size. Maximization of the second term is called a linear assignment problem, which can be solved efficiently by the Hungarian method [88].

The above method is called *least-squares object matching* (LSOM) [32]. A MATLAB® implementation of LSOM is publicly available [89].

3.7. Clustering

Here, we show how SMI can be effectively used for clustering.

3.7.1. Introduction

The objective of clustering is to classify data samples into disjoint groups in an unsupervised manner. K-means [90] is a classic but still popular clustering algorithm. However, k-means only produces linearly separated clusters, and thus its usefulness is rather limited in practice. To cope with this problem, various non-linear clustering methods have been developed. Kernel k-means [91] performs k-means in a feature space induced by a reproducing kernel function [46]. Spectral clustering [92,93] first unfolds non-linear data manifolds by a spectral embedding method, and then performs k-means in the embedded space. Blurring mean-shift [94,95] uses a non-parametric kernel density estimator for modeling the data-generating probability density, and finds clusters based on the modes of the estimated density. Discriminative clustering learns a discriminative classifier for separating clusters, where class labels are also treated as parameters to be optimized [96,97]. Dependence-maximization clustering determines cluster assignments so that their dependence on input data is maximized [34,98,99].

Information-maximization clustering exhibited the state-of-the-art performance [100,101], where probabilistic classifiers such as a kernelized Gaussian classifier [100] and a kernel logistic regression classifier [101] are learned so that mutual information between feature vectors and cluster assignments is maximized in an unsupervised manner. A notable advantage of information-maximization clustering is that classifier training is formulated as continuous optimization, which is substantially simpler than discrete optimization of cluster assignments. Indeed, classifier training can be carried out in computationally efficient manners by a gradient method [100] or a quasi-Newton method [101]. Furthermore, a model selection strategy based on the information-maximization principle is also provided [100]. Thus, kernel parameters can be systematically optimized in an unsupervised way. However, the optimization problems of these clustering methods are non-convex and finding a good local optimal solution is not straightforward in practice.

To overcome the above limitation, an SMI-based clustering method called *SMI clustering* (SMIC) was proposed [33]. Below, we review SMIC.

3.7.2. Clustering with SMI

Suppose that we are given d-dimensional i.i.d. feature vectors of size n, $\{\boldsymbol{x}_i \mid \boldsymbol{x}_i \in \mathbb{R}^d\}_{i=1}^n$, which are drawn independently from a distribution with density $p(\boldsymbol{x})$. The goal of clustering is to give cluster assignments, $\{y_i \mid y_i \in \{1,\ldots,c\}\}_{i=1}^n$, to the feature vectors $\{\boldsymbol{x}_i\}_{i=1}^n$, where c denotes the number of clusters. c is assumed to be pre-fixed below. To solve the clustering problem, the information-maximization approach is taken [100,101]. That is, clustering is regarded as an unsupervised classification problem, and the class-posterior probability $p(y|\boldsymbol{x})$ is learned so that "information" between feature vector $\boldsymbol{x}$ and cluster label y is maximized.

As an information measure, SMI Equation (1) is adopted, which can expressed as:

$$\mathrm{SMI} = \frac{1}{2}\int \sum_{y=1}^{c} p(y|\boldsymbol{x})p(\boldsymbol{x})\frac{p(y|\boldsymbol{x})}{p(y)}\mathrm{d}\boldsymbol{x} - \frac{1}{2} \tag{19}$$

Suppose that the class-prior probability $p(y)$ is set to a user-specified value π_y for $y = 1,\ldots,c$, where $\pi_y > 0$ and $\sum_{y=1}^{c}\pi_y = 1$. Without loss of generality, $\{\pi_y\}_{y=1}^{c}$ are assumed to be sorted in the ascending order:

$$\pi_1 \leq \cdots \leq \pi_c$$

If $\{\pi_y\}_{y=1}^{c}$ is unknown, the uniform class-prior distribution may be adopted:

$$p(y) = \frac{1}{c} \text{ for } y = 1,\ldots,c$$

Substituting π_y into $p(y)$, we can express Equation (19) as:

$$\frac{1}{2}\int \sum_{y=1}^{c} \frac{1}{\pi_y} p(y|\boldsymbol{x})p(\boldsymbol{x})p(y|\boldsymbol{x})\mathrm{d}\boldsymbol{x} - \frac{1}{2} \tag{20}$$

Let us approximate the class-posterior probability $p(y|\boldsymbol{x})$ by the following kernel model:

$$q_{\boldsymbol{\alpha}}(y|\boldsymbol{x}) := \sum_{i=1}^{n} \alpha_{y,i} K(\boldsymbol{x}, \boldsymbol{x}_i), \tag{21}$$

where $\boldsymbol{\alpha} = (\alpha_{1,1},\ldots,\alpha_{c,n})^\top$ is the parameter vector and $K(\boldsymbol{x}, \boldsymbol{x}')$ denotes a kernel function. A useful example of kernel functions is the local-scaling kernel [102] defined as:

$$K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \begin{cases} \exp\left(-\dfrac{\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2}{2\sigma_i\sigma_j}\right) & \text{if } \boldsymbol{x}_i \in \mathcal{N}_k(\boldsymbol{x}_j) \text{ or } \boldsymbol{x}_j \in \mathcal{N}_k(\boldsymbol{x}_i) \\ 0 & \text{otherwise} \end{cases}$$

where $\mathcal{N}_k(\boldsymbol{x})$ denotes the set of k nearest neighbors for $\boldsymbol{x}$ (k is the kernel parameter), σ_i is a local scaling factor defined as $\sigma_i = \|\boldsymbol{x}_i - \boldsymbol{x}_i^{(k)}\|$, and $\boldsymbol{x}_i^{(k)}$ is the k-th nearest neighbor of $\boldsymbol{x}_i$. Note that we did not include the normalization term in Equation (21) because model outputs will be normalized later (see Equation (22)).

Further approximating the expectation with respect to $p(\boldsymbol{x})$ included in Equation (20) by the empirical average of samples $\{\boldsymbol{x}_i\}_{i=1}^n$, we arrive at the following SMI approximator:

$$\widehat{\mathrm{SMI}} := \frac{1}{2n}\sum_{y=1}^{c}\frac{1}{\pi_y}\boldsymbol{\alpha}_y^\top \boldsymbol{K}^2\boldsymbol{\alpha}_y - \frac{1}{2}$$

where $\boldsymbol{\alpha}_y := (\alpha_{y,1},\ldots,\alpha_{y,n})^\top$ and $K_{i,j} := K(\boldsymbol{x}_i,\boldsymbol{x}_j)$.

For each cluster y, $\boldsymbol{\alpha}_y^\top \boldsymbol{K}^2\boldsymbol{\alpha}_y$ is maximized under $\|\boldsymbol{\alpha}_y\| = 1$. Since this is the Rayleigh quotient, the maximizer is given by the normalized principal eigenvector of $\boldsymbol{K}$ [104]. To avoid all the solutions $\{\boldsymbol{\alpha}_y\}_{y=1}^c$ to be reduced to the same principal eigenvector, their mutual orthogonality is imposed:

$$\boldsymbol{\alpha}_y^\top \boldsymbol{\alpha}_{y'} = 0 \quad \text{for } y \neq y'$$

Then the solutions are given by the normalized eigenvectors $\boldsymbol{\psi}_1,\ldots,\boldsymbol{\psi}_c$ associated with the eigenvalues $\lambda_1 \geq \cdots \geq \lambda_n \geq 0$ of $\boldsymbol{K}$. Since the sign of $\boldsymbol{\psi}_y$ is arbitrary, the sign is set as:

$$\widetilde{\boldsymbol{\psi}}_y = \boldsymbol{\psi}_y \times \mathrm{sign}(\boldsymbol{\psi}_y^\top \boldsymbol{1}_n)$$

where $\mathrm{sign}(\cdot)$ denotes the sign of a scalar and $\boldsymbol{1}_n$ denotes the n-dimensional vector with all ones.

On the other hand, because

$$p(y) = \int p(y|\boldsymbol{x})p(\boldsymbol{x})\mathrm{d}\boldsymbol{x} \approx \frac{1}{n}\sum_{i=1}^{n} q_{\boldsymbol{\alpha}}(y|\boldsymbol{x}_i) = \boldsymbol{\alpha}_y^\top \boldsymbol{K}\boldsymbol{1}_n$$

and the class-prior probability $p(y)$ was set to π_y for $y = 1,\ldots,c$, the following normalization condition is obtained:

$$\boldsymbol{\alpha}_y^\top \boldsymbol{K}\boldsymbol{1}_n = \pi_y \tag{22}$$

Furthermore, probability estimates should be non-negative, which can be achieved by rounding up negative outputs to zero.

By taking these normalization and non-negativity issues into account, cluster assignment y_i for $\boldsymbol{x}_i$ is determined as the maximizer of the approximation of $p(y|\boldsymbol{x}_i)$:

$$y_i = \operatorname*{argmax}_y \frac{[\max(\boldsymbol{0}_n, \boldsymbol{K}\widetilde{\boldsymbol{\psi}}_y)]_i}{\pi_y^{-1}\max(\boldsymbol{0}_n, \boldsymbol{K}\widetilde{\boldsymbol{\psi}}_y)^\top \boldsymbol{1}_n} = \operatorname*{argmax}_y \frac{\pi_y[\max(\boldsymbol{0}_n, \widetilde{\boldsymbol{\psi}}_y)]_i}{\max(\boldsymbol{0}_n, \widetilde{\boldsymbol{\psi}}_y)^\top \boldsymbol{1}_n}$$

where the "max" operation for vectors is applied in the element-wise manner and $[\cdot]_i$ denotes the i-th element of a vector. Note that $\boldsymbol{K}\widetilde{\boldsymbol{\psi}}_y = \lambda_y\widetilde{\boldsymbol{\psi}}_y$ was used in the above derivation. For out-of-sample prediction, cluster assignment y' for new sample $\boldsymbol{x}'$ may be obtained as:

$$y' := \operatorname*{argmax}_y \frac{\pi_y \max\left(0, \sum_{i=1}^n K(\boldsymbol{x}', \boldsymbol{x}_i)[\widetilde{\boldsymbol{\psi}}_y]_i\right)}{\lambda_y \max(\boldsymbol{0}_n, \widetilde{\boldsymbol{\psi}}_y)^\top \boldsymbol{1}_n}$$

The above method is called *SMI-based clustering* (SMIC) [33]. LSMI can be used for model selection of SMIC, *i.e.*, LSMI is computed as a function of the kernel parameter included in $K(\boldsymbol{x}, \boldsymbol{x}')$ and the maximizer of LSMI is chosen as the most promising one. A MATLAB® implementation of SMIC is publicly available [103].

3.8. Causal Direction Estimation

Finally, we show how the SMI estimator can be used for causal direction estimation.

3.8.1. Introduction

Learning causality from data is one of the important challenges in the artificial intelligence, statistics, and machine learning communities [105]. A traditional method of learning causal relationship from observational data is based on the linear-dependence Gaussian-noise model [106]. However, the linear-Gaussian assumption is too restrictive and may not be fulfilled in practice. Recently, non-Gaussianity and non-linearity have been shown to be beneficial in causal inference, because it can break symmetry between observed variables [107,108]. Since then, much attention has been paid to the discovery of non-linear causal relationship through non-Gaussian noise models [109].

In the framework of non-linear non-Gaussian causal inference, the relation between a cause X and an effect Y is assumed to be described by $Y = f(X) + E$, where f is a non-linear function and E is non-Gaussian additive noise that is independent of the cause X. Under this additive noise assumption, it was shown [108] that the causal direction between X and Y can be identified based on a hypothesis test of whether the causal model $Y = f(X) + E$ or the alternative model $X = f'(Y) + E'$ fits the data well—here, the goodness of fit is measured by independence between inputs and residuals (*i.e.*, estimated noise). In [108], the functions f and f' were learned by the Gaussian process (GP) regression [110], and the independence between inputs and residuals was evaluated by the Hilbert–Schmidt independence criterion (HSIC) [85].

However, standard regression methods such as GP are designed to handle Gaussian noise, and thus they may not be suited for discovering causality in the non-Gaussian additive noise formulation. To cope with this problem, an alternative regression method called HSIC regression was proposed [109], which learns a function so that the dependence between inputs and residuals is directly minimized based on HSIC. Through experiments, HSIC regression was shown to outperform the GP-based method [109]. However, the choice of the kernel width in HSIC regression heavily affects the sensitivity of the independence measure, and systematic model selection strategies are not available. Another weakness of HSIC regression is that the kernel width of the regression model is fixed to the same value as HSIC. This crucially limits the flexibility of function approximation in HSIC regression.

To overcome the above weaknesses, an SMI-based regression method for causal inference called *least-squares independence regression* (LSIR) was developed [35]. Below, we review LSIR.

3.8.2. Dependence Minimizing Regression with SMI

Suppose random variables $X \in \mathbb{R}$ and $Y \in \mathbb{R}$ are connected by the following additive noise model [108]:

$$Y = f(X) + E$$

where $f : \mathbb{R} \to \mathbb{R}$ is some non-linear function and $E \in \mathbb{R}$ is a zero-mean random variable that is independent of X. The goal of dependence minimizing regression is, from i.i.d. paired samples $\{(x_i, y_i)\}_{i=1}^n$, to obtain a function $\widehat{f}$ such that input X and estimated additive noise $\widehat{E} = Y - \widehat{f}(X)$ are independent.

Let us employ a linear model for dependence minimizing regression:

$$f_{\boldsymbol{\beta}}(x) = \sum_{l=1}^{m} \beta_l \psi_l(x) = \boldsymbol{\beta}^\top \boldsymbol{\psi}(x)$$

where m is the number of basis functions, $\boldsymbol{\beta} = (\beta_1, \dots, \beta_m)^\top$ are regression parameters, and $\boldsymbol{\psi}(x) = (\psi_1(x), \dots, \psi_m(x))^\top$ are basis functions. In LSMI-based dependence minimization regression, the regression parameters $\boldsymbol{\beta}$ are learned as:

$$\min_{\boldsymbol{\beta}} \left[\mathrm{LSMI}\big(\{(x_i, e_i)\}_{i=1}^n\big) + \frac{\gamma}{2} \boldsymbol{\beta}^\top \boldsymbol{\beta} \right]$$

where $e_i = y_i - f_{\boldsymbol{\beta}}(x_i)$ is the residual and $\gamma > 0$ is the regularization parameter to avoid overfitting.

For regression parameter learning, a gradient descent method may be used:

$$\boldsymbol{\beta} \longleftarrow \boldsymbol{\beta} - t \left(\frac{\partial \mathrm{LSMI}}{\partial \boldsymbol{\beta}} + \gamma \boldsymbol{\beta} \right)$$

where t is the step size. The gradient $\frac{\partial \mathrm{LSMI}}{\partial \boldsymbol{\beta}}$ can be approximately expressed as:

$$\frac{\partial \mathrm{LSMI}}{\partial \boldsymbol{\beta}} = \sum_{\ell=1}^{n} \widehat{\theta}_\ell \frac{\partial \widehat{h}_\ell}{\partial \boldsymbol{\beta}} - \frac{1}{2} \sum_{\ell,\ell'=1}^{n} \widehat{\theta}_\ell \widehat{\theta}_{\ell'} \frac{\partial \widehat{H}_{\ell,\ell'}}{\partial \boldsymbol{\beta}}$$

where

$$\begin{aligned}
\frac{\partial \widehat{h}_\ell}{\partial \boldsymbol{\beta}} &\approx -\frac{1}{2n\sigma^2} \sum_{j=1}^{n} \exp\left(-\frac{(x_i - x_\ell)^2 + (e_i - e_\ell)^2}{2\sigma^2} \right) (e_i - e_\ell) \boldsymbol{\psi}(x_i), \\
\frac{\partial \widehat{H}_{\ell,\ell'}}{\partial \boldsymbol{\beta}} &\approx -\frac{1}{2n^2\sigma^2} \sum_{i,j=1}^{n} \exp\left(-\frac{(x_i - x_\ell)^2 + (e_j - e_\ell)^2 + (x_i - x_{\ell'})^2 + (e_j - e_{\ell'})^2}{2\sigma^2} \right) \\
&\quad \times \Big((e_j - e_\ell) \boldsymbol{\psi}(x_i) + (e_i - e_\ell) \boldsymbol{\psi}(x_j) \Big)
\end{aligned}$$

Note that, in the above derivation, the dependence of $\boldsymbol{\beta}$ on e_i is ignored for simplicity. Although it is possible to exactly compute the derivative in principle, this approximated expression is computationally more efficient with good performance in practice.

By taking into account the assumption that the mean of noise E is zero, the final regressor is obtained as:

$$\widehat{f}(x) = f_{\widehat{\boldsymbol{\beta}}}(x) + \frac{1}{n}\sum_{i=1}^{n}\Big(y_i - f_{\widehat{\boldsymbol{\beta}}}(x_i)\Big)$$

This method is called *least-squares independence regression* (LSIR) [35]. A MATLAB® implementation of LSIR is publicly available [111].

3.8.3. Causal Direction Inference by LSIR

Our final goal is, given i.i.d. paired samples $\{(x_i, y_i)\}_{i=1}^{n}$, to determine whether X causes Y or vice versa under the additive noise assumption. To this end, we test whether the causal model $Y = f_Y(X) + E_Y$ or the alternative model $X = f_X(Y) + E_X$ fits the data well, where the goodness of fit is measured by independence between inputs and residuals (*i.e.*, estimated noise). Independence of inputs and residuals may be decided in practice based on the permutation test procedure [47].

More specifically, LSIR is first run for $\{(x_i, y_i)\}_{i=1}^{n}$ as usual, and obtain a regression function $\widehat{f}$. This procedure also provides an SMI estimate, $\mathrm{LSMI}(\{(x_i, \widehat{e}_i)\}_{i=1}^{n})$, where $\widehat{e}_i = y_i - \widehat{f}(x_i)$. Next, pairs of input and residual $\{(x_i, \widehat{e}_i)\}_{i=1}^{n}$ are randomly permuted as $\{(x_i, \widehat{e}_{\pi(i)})\}_{i=1}^{n}$, where $\pi(\cdot)$ is a randomly generated permutation function. Note that the permuted pairs of samples are independent of each other because the random permutation breaks the dependency between X and $\widehat{E}$ (if it exists). Then, an SMI estimate for the permuted data, $\mathrm{LSMI}(\{(x_i, \widehat{e}_{\pi(i)})\}_{i=1}^{n})$, is computed. This random permutation process is repeated many times, and the distribution of LSMI values under the null-hypothesis that X and $\widehat{E}$ are independent is constructed. Finally, the p-value is approximated by evaluating the relative ranking of LSMI computed from the original input-residual data, $\mathrm{LSMI}(\{(x_i, \widehat{e}_i)\}_{i=1}^{n})$, over the distribution of LSMI values for randomly permuted data.

In order to decide the causal direction, the p-values $p_{X\to Y}$ and $p_{X\leftarrow Y}$ for both directions $X \to Y$ (*i.e.*, X causes Y) and $X \leftarrow Y$ (*i.e.*, Y causes X) are computed. Then, for a given significance level δ, the causal direction is determined as follows:

- If $p_{X\to Y} > \delta$ and $p_{X\leftarrow Y} \leq \delta$, the causal model $X \to Y$ is chosen.
- If $p_{X\leftarrow Y} > \delta$ and $p_{X\to Y} \leq \delta$, the causal model $X \leftarrow Y$ is selected.
- If $p_{X\to Y}, p_{X\leftarrow Y} \leq \delta$, perhaps there is no causal relation between X and Y or our modeling assumption is not correct (e.g., an unobserved confounding variable exists).
- If $p_{X\to Y}, p_{X\leftarrow Y} > \delta$, perhaps our modeling assumption is not correct or it is not possible to identify a causal direction (*i.e.*, X, Y, and E are Gaussian random variables).

When we have prior knowledge that there exists a causal relation between X and Y but the causal direction is unknown, the values of $p_{X\to Y}$ and $p_{X\leftarrow Y}$ may be simply compared for determining the causal direction as follows:

- If $p_{X\to Y} > p_{X\leftarrow Y}$, we conclude that X causes Y.
- Otherwise, we conclude that Y causes X.

This simplified procedure does not include the computational expensive permutation process and thus it is computationally very efficient.

4. Conclusions

In this article, we reviewed recent development in the estimation of *squared-loss mutual information* (SMI) and its application to machine learning. The key idea for accurately estimating SMI is to directly estimate the ratio of probability densities without separately estimating each density. A notable advantage of the SMI estimator called *least-squares mutual information* (LSMI) [19] is that it can be computed analytically in a computationally more efficient and numerically more stable way than ordinary MI.

We have introduced SMI as a measure of statistical independence between random variables. On the other hand, ordinary MI has a rich information-theoretic interpretation via entropies. Thus, it is important to investigate an information-theoretic meaning of SMI, which remains to be an open question currently.

Various methods of direct density-ratio estimation have been explored so far [16,18], and such density ratio estimators were shown to be applicable to an even wider class of machine learning tasks beyond SMI estimation, such as non-stationarity adaptation [112], outlier detection [113], change detection [114,115], class-balance estimation [116], two-sample homogeneity testing [117,118], probabilistic classification [119,120], and conditional density estimation [121].

Improving the accuracy of density ratio estimation contributes to enhancing the performance of the above machine learning solutions. Recent advances in this line of research include dimensionality reduction for density ratio estimation [122–124], a unified statistical framework of density ratio estimation [18], and extensions to relative density ratios [125] and density differences [126]. Further improving the accuracy and computational efficiency and exploring new application areas are important future directions to pursue.

More program codes are publicly available [127].

Acknowledgements

The author would like to thank Taiji Suzuki, Makoto Yamada, Wittawat Jitkrittum, Masayuki Karasuyama, Marthinus Christoffel du Plessis, and John Quinn for their valuable comments. This work was supported by the JST PRESTO program, the FIRST program, and AOARD.

References

1. Shannon, C. A mathematical theory of communication. *AT&T Tech. J.* **1948**, *27*, 379–423.
2. Cover, T.M.; Thomas, J.A. *Elements of Information Theory*, 2nd ed.; John Wiley & Sons, Inc.: Hoboken, NJ, USA, 2006.
3. Kullback, S.; Leibler, R.A. On information and sufficiency. *Ann. Math. Stat.* **1951**, *22*, 79–86.
4. Fraser, A.M.; Swinney, H.L. Independent coordinates for strange attractors from mutual information. *Phys. Rev. A* **1986**, *33*, 1134–1140.
5. Vapnik, V.N. *Statistical Learning Theory*; Wiley: New York, NY, USA, 1998.
6. Darbellay, G.A.; Vajda, I. Estimation of the information by an adaptive partitioning of the observation space. *IEEE Trans. Inf. Theory* **1999**, *45*, 1315–1321.

7. Wang, Q.; Kulkarmi, S.R.; Verdú, S. Divergence estimation of continuous distributions based on data-dependent partitions. *IEEE Trans. Inf. Theory* **2005**, *51*, 3064–3074.
8. Silva, J.; Narayanan, S. Universal Consistency of Data-Driven Partitions for Divergence Estimation. In Proceedings of IEEE International Symposium on Information Theory, Nice, France, 24–29 June 2007; pp. 2021–2025.
9. Kraskov, A.; Stögbauer, H.; Grassberger, P. Estimating mutual information. *Phys. Rev. E* **2004**, *69*, 066138.
10. Khan, S.; Bandyopadhyay, S.; Ganguly, A.; Saigal, S. Relative performance of mutual information estimation methods for quantifying the dependence among short and noisy data. *Phys. Rev. E* **2007**, *76*, 026209.
11. Pérez-Cruz, F. Kullback-Leibler Divergence Estimation of Continuous Distributions. In Proceedings of IEEE International Symposium on Information Theory, Toronto, Canada, 6–11 July 2008; pp. 1666–1670.
12. Van Hulle, M.M. Edgeworth approximation of multivariate differential entropy. *Neural Comput.* **2005**, *17*, 1903–1910.
13. Suzuki, T.; Sugiyama, M.; Sese, J.; Kanamori, T. Approximating Mutual Information by Maximum Likelihood Density Ratio Estimation. In Proceedings of ECML-PKDD2008 Workshop on New Challenges for Feature Selection in Data Mining and Knowledge Discovery 2008 (FSDM2008); Saeys, Y., Liu, H., Inza, I., Wehenkel, L., de Peer, Y.V., Eds.; 2008; Volume 4, *JMLR Workshop and Conference Proceedings*, pp. 5–20.
14. Sugiyama, M.; Suzuki, T.; Nakajima, S.; Kashima, H.; von Bünau, P.; Kawanabe, M. Direct importance estimation for covariate shift adaptation. *Ann. I. Stat. Math.* **2008**, *60*, 699–746.
15. Nguyen, X.; Wainwright, M.J.; Jordan, M.I. Estimating divergence functionals and the likelihood ratio by convex risk minimization. *IEEE Trans. Inf. Theory* **2010**, *56*, 5847–5861.
16. Sugiyama, M.; Suzuki, T.; Kanamori, T. *Density Ratio Estimation in Machine Learning*; Cambridge University Press: Cambridge, UK, 2012.
17. Basu, A.; Harris, I.R.; Hjort, N.L.; Jones, M.C. Robust and efficient estimation by minimising a density power divergence. *Biometrika* **1998**, *85*, 549–559.
18. Sugiyama, M.; Suzuki, T.; Kanamori, T. Density ratio matching under the bregman divergence: A unified framework of density ratio estimation. *Ann. I. Stat. Math.* **2012**, *64*, 1009–1044.
19. Suzuki, T.; Sugiyama, M.; Kanamori, T.; Sese, J. Mutual information estimation reveals global associations between stimuli and biological processes. *BMC Bioinf.* **2009**, *10*, S52:1–S52:12.
20. Pearson, K. On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling. *Philos. Mag. Series 5* **1900**, *50*, 157–175.
21. Ali, S.M.; Silvey, S.D. A general class of coefficients of divergence of one distribution from another. *J. R. Stat. Soc. Series B* **1966**, *28*, 131–142.
22. Csiszár, I. Information-type measures of difference of probability distributions and indirect observation. *Stud. Sci. Math. Hung.* **1967**, *2*, 229–318.
23. Kanamori, T.; Hido, S.; Sugiyama, M. A least-squares approach to direct importance estimation. *J. Mach. Learn. Res.* **2009**, *10*, 1391–1445.

24. Kanamori, T.; Suzuki, T.; Sugiyama, M. Statistical Analysis of kernel-based least-squares density-ratio estimation. *Mach. Learn.* **2012**, *86*, 335–367.
25. Kanamori, T.; Suzuki, T.; Sugiyama, M. Computational complexity of kernel-based density-ratio estimation: A condition number analysis. **2009**, arXiv:0912.2800.
26. Sugiyama, M.; Suzuki, T. Least-squares independence test. *IEICE T. Inf. Syst.* **2011**, *E94-D*, 1333–1336.
27. Jitkrittum, W.; Hachiya, H.; Sugiyama, M. Feature Selection via ℓ_1-Penalized Squared-Loss Mutual Information. Technical Report 1210.1960, arXiv, 2012.
28. Suzuki, T.; Sugiyama, M. Sufficient dimension reduction via squared-loss mutual information estimation. Available online: sugiyama-www.cs.titech.ac.jp/.../AISTATS2010b.pdf (accessed on 26 December 2012).
29. Yamada, M.; Niu, G.; Takagi, J.; Sugiyama, M. Computationally Efficient Sufficient Dimension Reduction via Squared-Loss Mutual Information. In Proceedings of the Third Asian Conference on Machine Learning (ACML2011); Hsu, C.N., Lee, W.S., Eds.; 2011; Volume 20, *JMLR Workshop and Conference Proceedings*, pp. 247–262.
30. Karasuyama, M.; Sugiyama. Canonical dependency analysis based on squared-loss mutual information. *Neural Netw.* **2012**, *34*, 46–55.
31. Suzuki, T.; Sugiyama, M. Least-squares independent component analysis. *Neural Comput.* **2011**, *23*, 284–301.
32. Yamada, M.; Sugiyama, M. Cross-Domain Object Matching with Model Selection. In Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics (AISTATS2011); Gordon, G., Dunson, D.; Dudík, M., Eds.; 2011; Volume 15, *JMLR Workshop and Conference Proceedings*, pp. 807–815.
33. Sugiyama, M.; Yamada, M.; Kimura, M.; Hachiya, H. On Information-Maximization Clustering: Tuning Parameter Selection and Analytic Solution. In Proceedings of 28th International Conference on Machine Learning (ICML2011); Getoor, L., Scheffer, T., Eds.; 2011; pp. 65–72.
34. Kimura, M.; Sugiyama, M. Dependence-maximization clustering with least-squares mutual information. *J. Adv. Comput. Intell. Intell. Inf.* **2011**, *15*, 800–805.
35. Yamada, M.; Sugiyama, M. Dependence Minimizing Regression with Model Selection for Non-Linear Causal Inference under Non-Gaussian Noise. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence (AAAI2010)*; The AAAI Press: Atlanta, Georgia, USA, 2010; pp. 643–648.
36. Van der Vaart, A.W.; Wellner, J.A. *Weak Convergence and Empirical Processes with Applications to Statistics*; Springer: New York, NY, USA, 1996.
37. Van der Vaart, A.W. *Asymptotic Statistics*; Cambridge University Press: Cambridge, MA, USA, 2000.
38. Aronszajn, N. Theory of reproducing kernels. *T. Am. Math. Soc.* **1950**, *68*, 337–404.
39. Least-Squares Mutual Information (LSMI). Available online: http://sugiyama-www.cs.titech.ac.jp/~sugi/software/LSMI/ (accessed on 7 December 2012).
40. Efron, B.; Hastie, T.; Johnstone, I.; Tibshirani, R. Least angle regression. *Ann. Stat.* **2004**, *32*, 407–499.

41. Hastie, T.; Rosset, S.; Tibshirani, R.; Zhu, J. The entire regularization path for the support vector machine. *J. Mach. Learn. Res.* **2004**, *5*, 1391–1415.
42. Gärtner, T. A survey of kernels for structured data. *SIGKDD Explor.* **2003**, *5*, S268–S275.
43. Sarwar, B.; Karypis, G.; Konstan, J.; Reidl, J. Item-Based Collaborative Filtering Recommendation Algorithms. In Proceedings of the 10th International Conference on World Wide Web (WWW2001), Hong Kong, China, 1–5 May 2001; pp. 285–295.
44. Gretton, A.; Fukumizu, K.; Teo, C.H.; Song, L.; Schölkopf, B.; Smola, A. A Kernel Statistical Test of Independence. Advances in Neural Information Processing Systems 20; Platt, J.C., Koller, D., Singer, Y., Roweis, S., Eds.; MIT Press: Cambridge, MA, USA, 2008; pp. 585–592.
45. Steinwart, I. On the influence of the kernel on the consistency of support vector machines. *J. Mach. Learn. Res.* **2001**, *2*, 67–93.
46. Schölkopf, B.; Smola, A.J. *Learning with Kernels*; MIT Press: Cambridge, MA, USA, 2002.
47. Efron, B.; Tibshirani, R.J. *An Introduction to the Bootstrap*; Chapman & Hall/CRC: New York, NY, USA, 1993.
48. Least-Squares Independence Test (LSIT). Available online: http://sugiyama-www.cs.titech.ac.jp/~sugi/software/LSIT/ (accessed on 7 December 2012).
49. Guyon, I.; Elisseeff, A. An introduction to variable and feature selection. *J. Mach. Learn. Res.* **2003**, *3*, 1157–1182.
50. Tibshirani, R. Regression shrinkage and subset selection with the lasso. *J. R. Stat. Soc. Series B* **1996**, *58*, 267–288.
51. Boyd, S.; Vandenberghe, L. *Convex Optimization*; Cambridge University Press: Cambridge, UK, 2004.
52. Tomioka, R.; Suzuki, T.; Sugiyama, M. Super-linear convergence of dual augmented Lagrangian algorithm for sparsity regularized estimation. *J. Mach. Learn. Res.* **2011**, *12*, 1537–1586.
53. ℓ_1-Ball. Available online: http://wittawat.com/software/l1lsmi/ (accessed on 7 December).
54. Duchi, J.; Shalev-Shwartz, S.; Singer, Y.; Chandra, T. Efficient Projections onto the ℓ_1-Ball for Learning in High Dimensions. In Proceedings of the 25th Annual International Conference on Machine Learning (ICML2008); McCallum, A., Roweis, S., Eds.; Helsinki, Finland, 5–9 July 2008; pp. 272–279.
55. Cook, R.D. *Regression Graphics: Ideas for Studying Regressions through Graphics*; Wiley: New York, NY, USA, 1998.
56. Li, K. Sliced inverse regression for dimension reduction. *J. Am. Stat. Assoc.* **1991**, *86*, 316–342.
57. Li, K. On principal hessian directions for data visualization and dimension reduction: another application of Stein's lemma. *J. Am. Stat. Assoc.* **1992**, *87*, 1025–1039.
58. Cook, R.D. SAVE: A method for dimension reduction and graphics in regression. *Commun. Stat. Theory* **2000**, *29*, 2109–2121.
59. Fukumizu, K.; Bach, F.R.; Jordan, M.I. Kernel dimension reduction in regression. *Ann. Stat.* **2009**, *37*, 1871–1905.
60. Golub, G.H.; Loan, C.F.V. *Matrix Computations*, 2nd ed.; Johns Hopkins University Press: Baltimore, MD, USA, 1989.

61. Nishimori, Y.; Akaho, S. Learning algorithms utilizing quasi-geodesic flows on the Stiefel manifold. *Neurocomputing* **2005**, *67*, 106–135.
62. Amari, S. Natural gradient works efficiently in learning. *Neural Comput.* **1998**, *10*, 251–276.
63. Edelman, A.; Arias, T.A.; Smith, S.T. The geometry of algorithms with orthogonality constraints. *SIAM J. Matrix. Anal. A.* **1998**, *20*, 303–353.
64. Patriksson, M. *Nonlinear Programming and Variational Inequality Problems*; Kluwer Academic: Dordrecht, The Netherlands, 1999.
65. Least-Squares Dimensionality Reduction (LSDR). Available online: http://sugiyama-www.cs.titech.ac.jp/~sugi/software/LSDR/ (accessed on 7 December 2012).
66. Epanechnikov, V. Nonparametric estimates of a multivariate probability density. *Theor. Probab. Appl.* **1969**, *14*, 153–158.
67. Sufficient Component Analysis (SCA). Available online: http://sugiyama-www.cs.titech.ac.jp/~yamada/sca.html (accessed on 7 December 2012).
68. Hotelling, H. Relations between two sets of variates. *Biometrika* **1936**, *28*, 321–377.
69. Becker, S.; Hinton, G.E. A self-organizing neural network that discovers surfaces in random-dot stereograms. *Nature* **1992**, *355*, 161–163.
70. Fyfe, C.; Lai, P.L. Kernel and nonlinear canonical correlation analysis. *Int. J. Neural Syst.* **2000**, *10*, 365–377.
71. Akaho, S. A Kernel Method For Canonical Correlation Analysis. In Proceedings of the International Meeting of the Psychometric Society, Osaka, Japan, 15–19 July 2001.
72. Gestel, T.V.; Suykens, J.; Brabanter, J.D.; Moor, B.D.; Vandewalle, J. Kernel Canonical Correlation Analysis and Least Squares Support Vector Machines. In Proceedings of the International Conference on Artificial Neural Networks; Springer Berlin/Heidelberg, Germany, 2001; Volume 2130, *Lecture Notes in Computer Science*, pp. 384–389.
73. Breiman, L.; Friedman, J.H. Estimating optimal transformations for multiple regression and correlation. *J. Am. Stat. Assoc.* **1985**, *80*, 580–598.
74. Bach, F.; Jordan, M.I. Kernel independent component analysis. *J. Mach. Learn. Res.* **2002**, *3*, 1–48.
75. Yin, X. Canonical correlation analysis based on information theory. *J. Multivariate Anal.* **2004**, *91*, 161–176.
76. Härdle, W.; Müller, M.; Sperlich, S.; Werwatz, A. *Nonparametric and Semiparametric Models*; Springer: Berlin, Germany, 2004.
77. Least-Squares Canonical Dependency Analysis (LSCDA). Available online: http://www.bic.kyoto-u.ac.jp/pathway/krsym/software/LSCDA/index.html (accessed on 7 December 2012).
78. Hyvärinen, A.; Karhunen, J.; Oja, E. *Independent Component Analysis*; Wiley: New York, NY, USA, 2001.
79. Amari, S.; Cichocki, A.; Yang, H.H. A New Learning Algorithm for Blind Signal Separation. Advances in Neural Information Processing Systems 8; Touretzky, D.S., Mozer, M.C., Hasselmo, M.E., Eds.; The MIT Press: Cambridge, MA, USA, 1996; pp. 757–763.
80. Van Hulle, M.M. Sequential fixed-point ICA based on mutual information minimization. *Neural Comput.* **2008**, *20*, 1344–1365.

81. Jutten, C.; Herault, J. Blind separation of sources, part I: An adaptive algorithm based on neuromimetic architecture. *Signal Process.* **1991**, *24*, 1–10.
82. Hyvärinen, A. Fast and robust fixed-point algorithms for independent component analysis. *IEEE T. Neural Networ.* **1999**, *10*, 626.
83. Least-squares Independent Component Analysis. Available online: http://www.simplex.t.u-tokyo.ac.jp/~s-taiji/software/LICA/index.html (accessed on 7 December 2012).
84. Jebara, T. Kernelized Sorting, Permutation and Alignment for Minimum Volume PCA. In *Proceedings of the 17th Annual Conference on Learning Theory (COLT2004)*, Banff, Canada, 1–4 July 2004; pp. 609–623.
85. Gretton, A.; Bousquet, O.; Smola, A.; Schölkopf, B. Measuring Statistical Dependence with Hilbert-Schmidt Norms. In *Algorithmic Learning Theory*; Jain, S., Simon, H.U., Tomita, E., Eds.; Springer-Verlag: Berlin, Germany, 2005; Lecture Notes in Artificial Intelligence, pp. 63–77.
86. Quadrianto, N.; Smola, A.J.; Song, L.; Tuytelaars, T. Kernelized sorting. *IEEE Trans. Patt. Anal.* **2010**, *32*, 1809–1821.
87. Jagarlamudi, J.; Juarez, S.; Daumé III, H. Kernelized Sorting for Natural Language Processing. In Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence (AAAI2010), Atlanta, Georgia, USA, 11–15 July 2010; pp. 1020–1025.
88. Kuhn, H.W. The Hungarian method for the assignment problem. *Nav. Res. Logist. Q.* **1955**, *2*, 83–97.
89. Least-Squares Object Matching (LSOM). Available online: http://sugiyama-www.cs.titech.ac.jp/~yamada/lsom.html (accessed on 7 December 2012).
90. MacQueen, J.B. Some Methods for Classification and Analysis of Multivariate Observations. In *Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability*; University of California Press: Berkeley, CA, USA, 1967; Vol. 1, pp. 281–297.
91. Girolami, M. Mercer kernel-based clustering in feature space. *IEEE Trans. Neural Networ.* **2002**, *13*, 780–784.
92. Shi, J.; Malik, J. Normalized cuts and image segmentation. *IEEE Trans. Patt. Anal.* **2000**, *22*, 888–905.
93. Ng, A.Y.; Jordan, M.I.; Weiss, Y. On Spectral Clustering: Analysis and An Algorithm. Advances in Neural Information Processing Systems 14; Dietterich, T.G., Becker, S., Ghahramani, Z., Eds.; MIT Press: Cambridge, MA, USA, 2002; pp. 849–856.
94. Fukunaga, K.; Hostetler, L.D. The estimation of the gradient of a density function, with application in pattern recognition. *IEEE Trans. Inf. Theory* **1975**, *21*, 32–40.
95. Carreira-Perpiñán, M.A. Fast Nonparametric Clustering with Gaussian Blurring Mean-Shift. In Proceedings of 23rd International Conference on Machine Learning (ICML2006); Cohen, W., Moore, A., Eds.; Pittsburgh, Pennsylvania, USA, 25–29 June 2006; pp. 153–160.
96. Xu, L.; Neufeld, J.; Larson, B.; Schuurmans, D. Maximum Margin Clustering. Advances in Neural Information Processing Systems 17; Saul, L.K., Weiss, Y., Bottou, L., Eds.; MIT Press: Cambridge, MA, USA, 2005; pp. 1537–1544.

97. Bach, F.; Harchaoui, Z. DIFFRAC: A Discriminative and Flexible Framework for Clustering. Advances in Neural Information Processing Systems 20; Platt, J.C., Koller, D., Singer, Y., Roweis, S., Eds.; MIT Press: Cambridge, MA, USA, 2008; pp. 49–56.
98. Song, L.; Smola, A.; Gretton, A.; Borgwardt, K. A Dependence Maximization View of Clustering. In Proceedings of the 24th Annual International Conference on Machine Learning (ICML2007); Ghahramani, Z., Ed.; Corvallis, Oregon, USA, 20–24 June 2007; pp. 815–822.
99. Faivishevsky, L.; Goldberger, J. A Nonparametric Information Theoretic Clustering Algorithm. In Proceedings of 27th International Conference on Machine Learning (ICML2010); Joachims, A.T., Fürnkranz, J., Eds.; Haifa, Israel, 21–24 June 2010; pp. 351–358.
100. Agakov, F.; Barber, D. Kernelized Infomax Clustering. Advances in Neural Information Processing Systems 18; Weiss, Y., Schölkopf, B., Platt, J., Eds.; MIT Press: Cambridge, MA, USA, 2006; pp. 17–24.
101. Gomes, R.; Krause, A.; Perona, P. Discriminative Clustering by Regularized Information Maximization. Advances in Neural Information Processing Systems 23; Lafferty, J., Williams, C.K.I., Zemel, R., Shawe-Taylor, J., Culotta, A., Eds.; 2010; pp. 766–774.
102. Zelnik-Manor, L.; Perona, P. Self-Tuning Spectral Clustering. Advances in Neural Information Processing Systems 17; Saul, L.K., Weiss, Y., Bottou, L., Eds.; MIT Press: Cambridge, MA, USA, 2005; pp. 1601–1608.
103. SMI-based Clustering (SMIC). Available online: http://sugiyama-www.cs.titech.ac.jp/~sugi/software/SMIC/ (accessed on 7 December 2012).
104. Horn, R.A.; Johnson, C.A. *Matrix Analysis*; Cambridge University Press: Cambridge, UK, 1985.
105. Pearl, J. *Causality: Models, Reasoning and Inference*; Cambridge University Press: New York, NY, USA, 2000.
106. Geiger, D.; Heckerman, D. Learning Gaussian Networks. In Proceedings of the 10th Annual Conference on Uncertainty in Artificial Intelligence (UAI1994), Seattle, Washington, USA, 29–31 July 1994; pp. 235–243.
107. Shimizu, S.; Hoyer, P.O.; Hyvärinen, A.; Kerminen, A.J. A linear non-gaussian acyclic model for causal discovery. *J. Mach. Learn. Res.* **2006**, *7*, 2003–2030.
108. Hoyer, P.O.; Janzing, D.; Mooij, J.M.; Peters, J.; Schölkopf, B. Nonlinear Causal Discovery with Additive Noise Models. Advances in Neural Information Processing Systems 21; Koller, D., Schuurmans, D., Bengio, Y., Bottou, L., Eds.; MIT Press: Cambridge, MA, USA, 2009; pp. 689–696.
109. Mooij, J.; Janzing, D.; Peters, J.; Schölkopf, B. Regression by Dependence Minimization and Its Application to Causal Inference in Additive Noise Models. In Proceedings of the 26th Annual International Conference on Machine Learning (ICML2009), Montreal, Canada Jun. 14–18, 2009; pp. 745–752.
110. Rasmussen, C.E.; Williams, C.K.I. *Gaussian Processes for Machine Learning*; MIT Press: Cambridge, MA, USA, 2006.
111. Least-Squares Independence Regression (LSIR). Availble online: http://sugiyama-www.cs.titech.ac.jp/~yamada/lsir.html (accessed on 7 December 2012).

112. Sugiyama, M.; Kawanabe, M. *Machine Learning in Non-Stationary Environments: Introduction to Covariate Shift Adaptation*; MIT Press: Cambridge, Massachusetts, USA, 2012.
113. Hido, S.; Tsuboi, Y.; Kashima, H.; Sugiyama, M.; Kanamori, T. Statistical outlier detection using direct density ratio estimation. *Knowl. Inf. Syst.* **2011**, *26*, 309–336.
114. Kawahara, Y.; Sugiyama, M. Sequential change-point detection based on direct density-ratio estimation. *Stat. Anal. Data Min.* **2012**, *5*, 114–127.
115. Liu, S.; Yamada, M.; Collier, N.; Sugiyama, M. Change-Point Detection in Time-Series Data by Relative Density-Ratio Estimation. In *Structural, Syntactic, and Statistical Pattern Recognition*; Gimel'farb, G., Hancock, E., Imiya, A., Kuijper, A., Kudo, M., Omachi, S., Windeatt, T., Yamada, K., Eds.; Springer: Berlin, Germany, 2012; Volume 7626, *Lecture Notes in Computer Science*, pp. 363–372.
116. Du Plessis, M.C.; Sugiyama, M. Semi-Supervised Learning of Class Balance under Class-Prior Change by Distribution Matching. In Proceedings of 29th International Conference on Machine Learning (ICML2012); Langford, J., Pineau, J., Eds.; Edinburgh, Scotland, 26 June–1 July 2012; pp. 823–830.
117. Sugiyama, M.; Suzuki, T.; Itoh, Y.; Kanamori, T.; Kimura, M. Least-squares two-sample test. *Neural Netw.* **2011**, *24*, 735–751.
118. Kanamori, T.; Suzuki, T.; Sugiyama, M. f-divergence estimation and two-sample homogeneity test under semiparametric density-ratio models. *IEEE Trans. Inf. Theory* **2012**, *58*, 708–720.
119. Sugiyama, M. Superfast-trainable multi-class probabilistic classifier by least-squares posterior fitting. *IEICE Trans. Inf. Syst.* **2010**, *E93-D*, 2690–2701.
120. Sugiyama, M.; Hachiya, H.; Yamada, M.; Simm, J.; Nam, H. Least-Squares Probabilistic Classifier: A Computationally Efficient Alternative to Kernel Logistic Regression. In Proceedings of International Workshop on Statistical Machine Learning for Speech Processing (IWSML2012), Kyoto, Japan, Mar. 31, 2012; pp. 1–10.
121. Sugiyama, M.; Takeuchi, I.; Suzuki, T.; Kanamori, T.; Hachiya, H.; Okanohara, D. Least-squares conditional density estimation. *IEICE Trans. Inf. Syst.* **2010**, *E93-D*, 583–594.
122. Sugiyama, M.; Kawanabe, M.; Chui, P.L. Dimensionality reduction for density ratio estimation in high-dimensional spaces. *Neural Netw.* **2010**, *23*, 44–59.
123. Sugiyama, M.; Yamada, M.; von Bünau, P.; Suzuki, T.; Kanamori, T.; Kawanabe, M. Direct density-ratio estimation with dimensionality reduction via least-squares hetero-distributional subspace search. *Neural Netw.* **2011**, *24*, 183–198.
124. Yamada, M.; Sugiyama, M. Direct Density-Ratio Estimation with Dimensionality Reduction via Hetero-Distributional Subspace Analysis. In Proceedings of the Twenty-Fifth AAAI Conference on Artificial Intelligence (AAAI2011); The AAAI Press: San Francisco, California, USA, 2011; pp. 549–554.
125. Yamada, M.; Suzuki, T.; Kanamori, T.; Hachiya, H.; Sugiyama, M. Relative Density-Ratio Estimation for Robust Distribution Comparison. Advances in Neural Information Processing Systems 24; Shawe-Taylor, J., Zemel, R.S., Bartlett, P., Pereira, F.C.N., Weinberger, K.Q., Eds.; 2011; pp. 594–602.

126. Sugiyama, M.; Suzuki, T.; Kanamori, T.; Du Plessis, M.C.; Liu, S.; Takeuchi, I. Density-Difference Estimation. Advances in Neural Information Processing Systems 25, 2012.
127. Software. Available online: http://sugiyama-www.cs.titech.ac.jp/~sugi/software/ (accessed on 7 December 2012).

Relative Density-Ratio Estimation for Robust Distribution Comparison

Makoto Yamada
NTT Communication Science Laboratories, NTT Corporation, Japan.
yamada.makoto@lab.ntt.co.jp

Taiji Suzuki
The University of Tokyo, Japan.
s-taiji@stat.t.u-tokyo.ac.jp

Takafumi Kanamori
Nagoya University, Japan.
kanamori@is.nagoya-u.ac.jp

Hirotaka Hachiya
Tokyo Institute of Technology, Japan.
hacchan@gmail.com

Masashi Sugiyama
Tokyo Institute of Technology, Japan.
sugi@cs.titech.ac.jp
http://sugiyama-www.cs.titech.ac.jp/˜sugi

Abstract

Divergence estimators based on direct approximation of density-ratios without going through separate approximation of numerator and denominator densities have been successfully applied to machine learning tasks that involve distribution comparison such as outlier detection, transfer learning, and two-sample homogeneity test. However, since density-ratio functions often possess high fluctuation, divergence estimation is still a challenging task in practice. In this paper, we propose to use *relative divergences* for distribution comparison, which involves approximation of *relative density-ratios*. Since relative density-ratios are always smoother than corresponding ordinary density-ratios, our proposed method is favorable in terms of the non-parametric convergence speed. Furthermore, we show that the proposed divergence estimator has asymptotic variance *independent* of the model complexity under a parametric setup, implying that the proposed estimator hardly overfits even with complex models. Through experiments, we demonstrate the usefulness of the proposed approach.

1 Introduction

Comparing probability distributions is a fundamental task in statistical data processing. It can be used for, e.g., *outlier detection* (Smola et al., 2009; Hido et al., 2011), *two-sample homogeneity test* (Gretton et al., 2007; Sugiyama et al., 2011), and *transfer learning* (Shimodaira, 2000; Sugiyama et al., 2007).

A standard approach to comparing probability densities $p(\boldsymbol{x})$ and $p'(\boldsymbol{x})$ would be to estimate a divergence from $p(\boldsymbol{x})$ to $p'(\boldsymbol{x})$, such as the *Kullback-Leibler (KL) divergence* (Kullback and Leibler, 1951):

$$\mathrm{KL}[p(\boldsymbol{x}), p'(\boldsymbol{x})] := \int \log\left(\frac{p(\boldsymbol{x})}{p'(\boldsymbol{x})}\right) p(\boldsymbol{x})\mathrm{d}\boldsymbol{x}.$$

A naive way to estimate the KL divergence is to separately approximate the densities $p(\boldsymbol{x})$ and $p'(\boldsymbol{x})$ from data and plug the estimated densities in the above definition. However, since density estimation is known to be a hard task (Vapnik, 1998), this approach does not work well unless a good parametric model is available. Recently, a divergence estimation approach which directly approximates the *density ratio*,

$$r(\boldsymbol{x}) := \frac{p(\boldsymbol{x})}{p'(\boldsymbol{x})},$$

without going through separate approximation of densities $p(\boldsymbol{x})$ and $p'(\boldsymbol{x})$ has been proposed (Sugiyama et al., 2008; Nguyen et al., 2010). Such density-ratio approximation methods were proved to achieve the optimal non-parametric convergence rate in the minimax sense.

However, the KL divergence estimation via density-ratio approximation is computationally rather expensive due to the non-linearity introduced by the 'log' term. To cope with this problem, another divergence called the *Pearson (PE) divergence* (Pearson, 1900) is useful. The PE divergence from $p(\boldsymbol{x})$ to $p'(\boldsymbol{x})$ is defined as

$$\mathrm{PE}[p(\boldsymbol{x}), p'(\boldsymbol{x})] := \frac{1}{2}\int \left(\frac{p(\boldsymbol{x})}{p'(\boldsymbol{x})} - 1\right)^2 p'(\boldsymbol{x})\mathrm{d}\boldsymbol{x}.$$

The PE divergence is a squared-loss variant of the KL divergence, and they both belong to the class of the *Ali-Silvey-Csiszár divergences* (which is also known as the *f-divergences*, see Ali and Silvey, 1966; Csiszár, 1967). Thus, the PE and KL divergences share similar properties, e.g., they are non-negative and vanish if and only if $p(\boldsymbol{x}) = p'(\boldsymbol{x})$.

Similarly to the KL divergence estimation, the PE divergence can also be accurately estimated based on density-ratio approximation (Kanamori et al., 2009): the density-ratio approximator called *unconstrained least-squares importance fitting* (uLSIF) gives the PE divergence estimator *analytically*, which can be computed just by solving a system of linear equations. The practical usefulness of the uLSIF-based PE divergence estimator was demonstrated in various applications such as outlier detection (Hido et al., 2011), two-sample homogeneity test (Sugiyama et al., 2011), and dimensionality reduction (Suzuki and Sugiyama, 2010).

In this paper, we first establish the non-parametric convergence rate of the uLSIF-based PE divergence estimator, which elucidates its superior theoretical properties. However, it also reveals that its convergence rate is actually governed by the 'sup'-norm of the true density-ratio function: $\max_{\boldsymbol{x}} r(\boldsymbol{x})$. This implies that, in the region where the denominator density $p'(\boldsymbol{x})$ takes small values, the density ratio $r(\boldsymbol{x}) = p(\boldsymbol{x})/p'(\boldsymbol{x})$ tends to take large values and therefore the overall convergence speed becomes slow. More critically, density ratios can even diverge to infinity under a rather simple setting, e.g., when the ratio of two Gaussian functions is considered (Cortes et al., 2010). This makes the paradigm of divergence estimation based on density-ratio approximation unreliable.

In order to overcome this fundamental problem, we propose an alternative approach to distribution comparison called *α-relative divergence estimation*. In the proposed approach, we estimate the quantity called the *α-relative divergence*, which is the divergence from $p(\boldsymbol{x})$ to the *α-mixture density* $\alpha p(\boldsymbol{x}) + (1-\alpha)p'(\boldsymbol{x})$ for $0 \leq \alpha < 1$. For example, the α-relative PE divergence is given by

$$\begin{aligned}\mathrm{PE}_\alpha[p(\boldsymbol{x}), p'(\boldsymbol{x})] &:= \mathrm{PE}[p(\boldsymbol{x}), \alpha p(\boldsymbol{x}) + (1-\alpha)p'(\boldsymbol{x})] \\ &= \frac{1}{2}\int \left(\frac{p(\boldsymbol{x})}{\alpha p(\boldsymbol{x}) + (1-\alpha)p'(\boldsymbol{x})} - 1\right)^2 (\alpha p(\boldsymbol{x}) + (1-\alpha)p'(\boldsymbol{x}))\,\mathrm{d}\boldsymbol{x}.\end{aligned}$$

We estimate the α-relative divergence by direct approximation of the *α-relative density-ratio*:

$$r_\alpha(\boldsymbol{x}) := \frac{p(\boldsymbol{x})}{\alpha p(\boldsymbol{x}) + (1-\alpha)p'(\boldsymbol{x})}.$$

A notable advantage of this approach is that the α-relative density-ratio is always bounded above by $1/\alpha$ when $\alpha > 0$, even when the ordinary density-ratio is unbounded. Based on this feature, we theoretically show that the α-relative PE divergence estimator based on α-relative density-ratio approximation is more favorable than the ordinary density-ratio approach in terms of the non-parametric convergence speed.

We further prove that, under a correctly-specified parametric setup, the asymptotic variance of our α-relative PE divergence estimator does not depend on the model complexity. This means that the proposed α-relative PE divergence estimator hardly overfits even with complex models.

Through extensive experiments on outlier detection, two-sample homogeneity test, and transfer learning, we demonstrate that our proposed α-relative PE divergence estimator compares favorably with alternative approaches.

The rest of this paper is structured as follows. In Section 2, our proposed relative PE divergence estimator is described. In Section 3, we provide non-parametric analysis of the convergence rate and parametric analysis of the variance of the proposed PE divergence estimator. In Section 4, we experimentally evaluate the performance of the proposed method on various tasks. Finally, in Section 5, we conclude the paper by summarizing our contributions and describing future prospects.

2 Estimation of Relative Pearson Divergence via Least-Squares Relative Density-Ratio Approximation

In this section, we propose an estimator of the relative Pearson (PE) divergence based on least-squares relative density-ratio approximation.

2.1 Problem Formulation

Suppose we are given independent and identically distributed (i.i.d.) samples $\{\boldsymbol{x}_i\}_{i=1}^{n}$ from a d-dimensional distribution P with density $p(\boldsymbol{x})$ and i.i.d. samples $\{\boldsymbol{x}'_j\}_{j=1}^{n'}$ from another d-dimensional distribution P' with density $p'(\boldsymbol{x})$:

$$\{\boldsymbol{x}_i\}_{i=1}^{n} \overset{\text{i.i.d.}}{\sim} P,$$
$$\{\boldsymbol{x}'_j\}_{j=1}^{n'} \overset{\text{i.i.d.}}{\sim} P'.$$

The goal of this paper is to compare the two underlying distributions P and P' only using the two sets of samples $\{\boldsymbol{x}_i\}_{i=1}^{n}$ and $\{\boldsymbol{x}'_j\}_{j=1}^{n'}$.

For $0 \leq \alpha < 1$, let $q_\alpha(\boldsymbol{x})$ be the α-*mixture density* of $p(\boldsymbol{x})$ and $p'(\boldsymbol{x})$:

$$q_\alpha(\boldsymbol{x}) := \alpha p(\boldsymbol{x}) + (1-\alpha)p'(\boldsymbol{x}).$$

Let $r_\alpha(\boldsymbol{x})$ be the α-*relative density-ratio* of $p(\boldsymbol{x})$ and $p'(\boldsymbol{x})$:

$$r_\alpha(\boldsymbol{x}) := \frac{p(\boldsymbol{x})}{\alpha p(\boldsymbol{x}) + (1-\alpha)p'(\boldsymbol{x})} = \frac{p(\boldsymbol{x})}{q_\alpha(\boldsymbol{x})}. \tag{1}$$

We define *the* α-*relative PE divergence* from $p(\boldsymbol{x})$ to $p'(\boldsymbol{x})$ as

$$\mathrm{PE}_\alpha := \frac{1}{2}\mathbb{E}_{q_\alpha(\boldsymbol{x})}\left[(r_\alpha(\boldsymbol{x}) - 1)^2\right], \tag{2}$$

where $\mathbb{E}_{p(\boldsymbol{x})}[f(\boldsymbol{x})]$ denotes the expectation of $f(\boldsymbol{x})$ under $p(\boldsymbol{x})$:

$$\mathbb{E}_{p(\boldsymbol{x})}[f(\boldsymbol{x})] = \int f(\boldsymbol{x})p(\boldsymbol{x})\mathrm{d}\boldsymbol{x}.$$

When $\alpha = 0$, PE_α is reduced to the ordinary PE divergence. Thus, the α-relative PE divergence can be regarded as a ‘smoothed’ extension of the ordinary PE divergence.

Below, we give a method for estimating the α-relative PE divergence based on the approximation of the α-relative density-ratio.

2.2 Direct Approximation of α-Relative Density-Ratios

Here, we describe a method for approximating the α-relative density-ratio (1).

Let us model the α-relative density-ratio $r_\alpha(\boldsymbol{x})$ by the following kernel model:

$$g(\boldsymbol{x};\boldsymbol{\theta}) := \sum_{\ell=1}^{n} \theta_\ell K(\boldsymbol{x}, \boldsymbol{x}_\ell),$$

where $\boldsymbol{\theta} := (\theta_1, \dots, \theta_n)^\top$ are parameters to be learned from data samples, $^\top$ denotes the transpose of a matrix or a vector, and $K(\boldsymbol{x}, \boldsymbol{x}')$ is a kernel basis function. In the experiments, we use the Gaussian kernel:

$$K(\boldsymbol{x}, \boldsymbol{x}') = \exp\left(-\frac{\|\boldsymbol{x}-\boldsymbol{x}'\|^2}{2\sigma^2}\right), \tag{3}$$

where σ (> 0) is the kernel width.

The parameters $\boldsymbol{\theta}$ in the model $g(\boldsymbol{x};\boldsymbol{\theta})$ are determined so that the following expected squared-error J is minimized:

$$\begin{aligned} J(\boldsymbol{\theta}) &:= \frac{1}{2}\mathbb{E}_{q_\alpha(\boldsymbol{x})}\left[\left(g(\boldsymbol{x};\boldsymbol{\theta}) - r_\alpha(\boldsymbol{x})\right)^2\right] \\ &= \frac{\alpha}{2}\mathbb{E}_{p(\boldsymbol{x})}\left[g(\boldsymbol{x};\boldsymbol{\theta})^2\right] + \frac{(1-\alpha)}{2}\mathbb{E}_{p'(\boldsymbol{x})}\left[g(\boldsymbol{x};\boldsymbol{\theta})^2\right] - \mathbb{E}_{p(\boldsymbol{x})}\left[g(\boldsymbol{x};\boldsymbol{\theta})\right] + \text{Const.}, \end{aligned}$$

where we used $r_\alpha(\boldsymbol{x})q_\alpha(\boldsymbol{x}) = p(\boldsymbol{x})$ in the third term. Approximating the expectations by empirical averages, we obtain the following optimization problem:

$$\widehat{\boldsymbol{\theta}} := \mathop{\mathrm{argmin}}_{\boldsymbol{\theta}\in\mathbb{R}^n} \left[\frac{1}{2}\boldsymbol{\theta}^\top \widehat{\boldsymbol{H}}\boldsymbol{\theta} - \widehat{\boldsymbol{h}}^\top \boldsymbol{\theta} + \frac{\lambda}{2}\boldsymbol{\theta}^\top\boldsymbol{\theta}\right], \tag{4}$$

where a penalty term $\lambda\boldsymbol{\theta}^\top\boldsymbol{\theta}/2$ is included for regularization purposes, and λ (≥ 0) denotes the regularization parameter. $\widehat{\boldsymbol{H}}$ is the $n \times n$ matrix with the (ℓ, ℓ')-th element

$$\widehat{H}_{\ell,\ell'} := \frac{\alpha}{n}\sum_{i=1}^{n} K(\boldsymbol{x}_i, \boldsymbol{x}_\ell)K(\boldsymbol{x}_i, \boldsymbol{x}_{\ell'}) + \frac{(1-\alpha)}{n'}\sum_{j=1}^{n'} K(\boldsymbol{x}'_j, \boldsymbol{x}_\ell)K(\boldsymbol{x}'_j, \boldsymbol{x}_{\ell'}). \tag{5}$$

$\widehat{\boldsymbol{h}}$ is the n-dimensional vector with the ℓ-th element

$$\widehat{h}_\ell := \frac{1}{n}\sum_{i=1}^{n} K(\boldsymbol{x}_i, \boldsymbol{x}_\ell).$$

It is easy to confirm that the solution of Eq.(4) can be *analytically* obtained as

$$\widehat{\boldsymbol{\theta}} = (\widehat{\boldsymbol{H}} + \lambda \boldsymbol{I}_n)^{-1}\widehat{\boldsymbol{h}},$$

where $\boldsymbol{I}_n$ denotes the n-dimensional identity matrix. Finally, a density-ratio estimator is given as

$$\widehat{r}_\alpha(\boldsymbol{x}) := g(\boldsymbol{x}; \widehat{\boldsymbol{\theta}}) = \sum_{\ell=1}^{n} \widehat{\theta}_\ell K(\boldsymbol{x}, \boldsymbol{x}_\ell). \tag{6}$$

When $\alpha = 0$, the above method is reduced to a direct density-ratio estimator called *unconstrained least-squares importance fitting* (uLSIF; Kanamori et al., 2009). Thus, the above method can be regarded as an extension of uLSIF to the α-relative density-ratio. For this reason, we refer to our method as *relative uLSIF* (RuLSIF).

The performance of RuLSIF depends on the choice of the kernel function (the kernel width σ in the case of the Gaussian kernel) and the regularization parameter λ. Model selection of RuLSIF is possible based on cross-validation with respect to the squared-error criterion J, in the same way as the original uLSIF (Kanamori et al., 2009).

2.3 α-Relative PE Divergence Estimation Based on RuLSIF

Using an estimator of the α-relative density-ratio $r_\alpha(\boldsymbol{x})$, we can construct estimators of the α-relative PE divergence (2). After a few lines of calculation, we can show that the α-relative PE divergence (2) is equivalently expressed as

$$\begin{aligned}\mathrm{PE}_\alpha &= -\frac{\alpha}{2}\mathbb{E}_{p(\boldsymbol{x})}\left[r_\alpha(\boldsymbol{x})^2\right] - \frac{(1-\alpha)}{2}\mathbb{E}_{p'(\boldsymbol{x})}\left[r_\alpha(\boldsymbol{x})^2\right] + \mathbb{E}_{p(\boldsymbol{x})}\left[r_\alpha(\boldsymbol{x})\right] - \frac{1}{2}\\ &= \frac{1}{2}\mathbb{E}_{p(\boldsymbol{x})}\left[r_\alpha(\boldsymbol{x})\right] - \frac{1}{2}.\end{aligned}$$

Note that the first line can also be obtained via *Legendre-Fenchel convex duality* of the divergence functional (Rockafellar, 1970).

Based on these expressions, we consider the following two estimators:

$$\widehat{\mathrm{PE}}_\alpha := -\frac{\alpha}{2n}\sum_{i=1}^{n}\widehat{r}(\boldsymbol{x}_i)^2 - \frac{(1-\alpha)}{2n'}\sum_{j=1}^{n'}\widehat{r}(\boldsymbol{x}'_j)^2 + \frac{1}{n}\sum_{i=1}^{n}\widehat{r}(\boldsymbol{x}_i) - \frac{1}{2}, \tag{7}$$

$$\widetilde{\mathrm{PE}}_\alpha := \frac{1}{2n}\sum_{i=1}^{n}\widehat{r}(\boldsymbol{x}_i) - \frac{1}{2}. \tag{8}$$

We note that the α-relative PE divergence (2) can have further different expressions than the above ones, and corresponding estimators can also be constructed similarly. However, the above two expressions will be particularly useful: the first estimator $\widehat{\mathrm{PE}}_\alpha$ has superior theoretical properties (see Section 3) and the second one $\widetilde{\mathrm{PE}}_\alpha$ is simple to compute.

2.4 Illustrative Examples

Here, we numerically illustrate the behavior of RuLSIF (6) using toy datasets. Let the numerator distribution be $P = N(0, 1)$, where $N(\mu, \sigma^2)$ denotes the normal distribution with mean μ and variance σ^2. The denominator distribution P' is set as follows:

(a) $P' = N(0, 1)$: P and P' are the same.

(b) $P' = N(0, 0.6)$: P' has smaller standard deviation than P.

(c) $P' = N(0, 2)$: P' has larger standard deviation than P.

(d) $P' = N(0.5, 1)$: P and P' have different means.

(e) $P' = 0.95N(0, 1) + 0.05N(3, 1)$: P' contains an additional component to P.

We draw $n = n' = 300$ samples from the above densities, and compute RuLSIF for $\alpha = 0$, 0.5, and 0.95.

Figure 1 shows the true densities, true density-ratios, and their estimates by RuLSIF. As can be seen from the graphs, the profiles of the true α-relative density-ratios get smoother as α increases. In particular, in the datasets (b) and (d), the true density-ratios for $\alpha = 0$ diverge to infinity, while those for $\alpha = 0.5$ and 0.95 are bounded (by $1/\alpha$). Overall, as α gets large, the estimation quality of RuLSIF tends to be improved since the complexity of true density-ratio functions is reduced.

Note that, in the dataset (a) where $p(\boldsymbol{x}) = p'(\boldsymbol{x})$, the true density-ratio $r_\alpha(\boldsymbol{x})$ does not depend on α since $r_\alpha(\boldsymbol{x}) = 1$ for any α. However, the estimated density-ratios still depend on α through the matrix $\widehat{\boldsymbol{H}}$ (see Eq.(5)).

3 Theoretical Analysis

In this section, we analyze theoretical properties of the proposed PE divergence estimators. More specifically, we provide non-parametric analysis of the convergence rate in Section 3.1, and parametric analysis of the estimation variance in Section 3.2. Since our theoretical analysis is highly technical, we focus on explaining practical insights we can gain from the theoretical results here; we describe all the mathematical details of the non-parametric convergence-rate analysis in Appendix A and the parametric variance analysis in Appendix B.

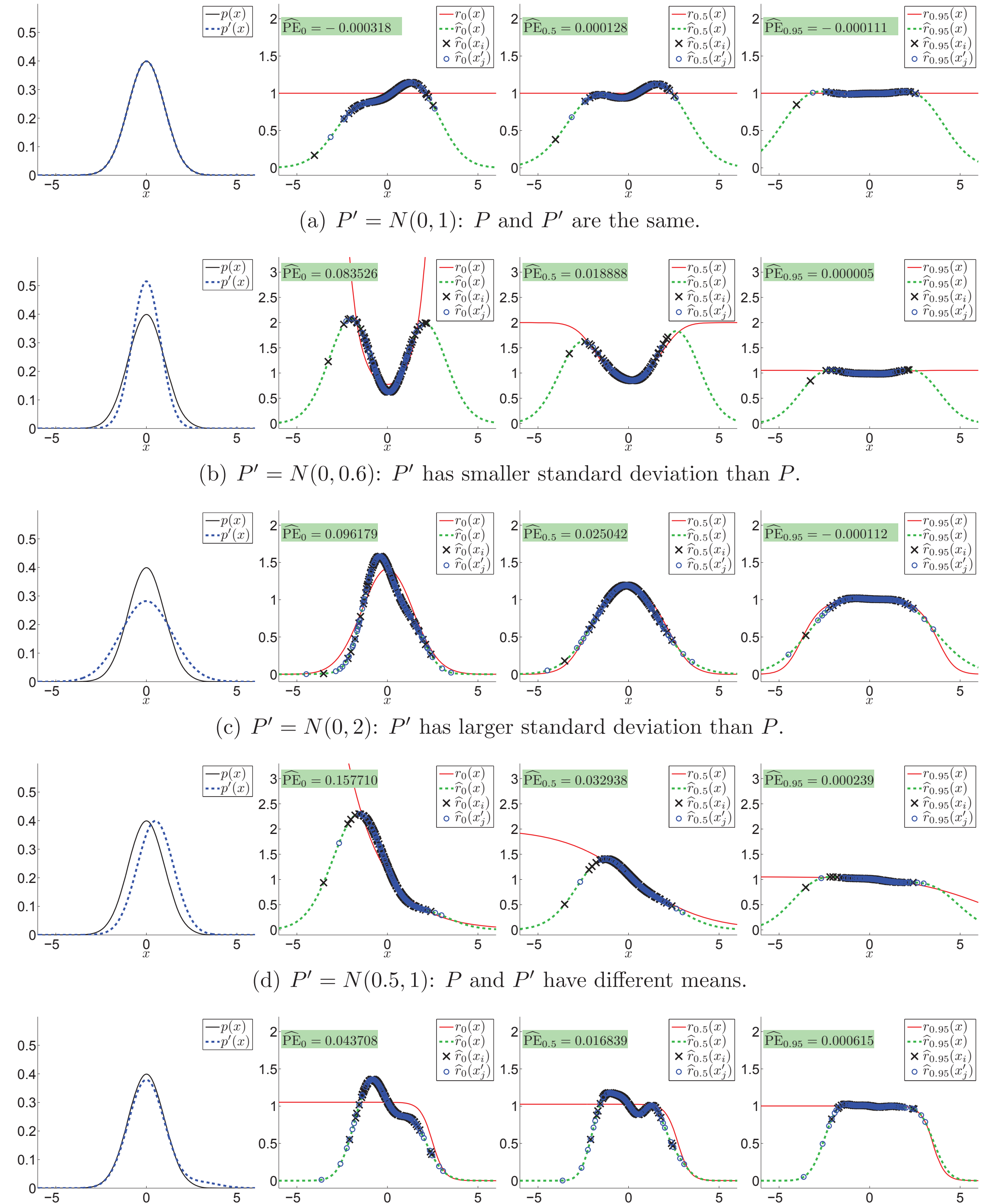
For theoretical analysis, let us consider a rather abstract form of our relative density-ratio estimator described as

$$\operatorname*{argmin}_{g \in \mathcal{G}} \left[\frac{\alpha}{2n} \sum_{i=1}^{n} g(\boldsymbol{x}_i)^2 + \frac{(1-\alpha)}{2n'} \sum_{j=1}^{n'} g(\boldsymbol{x}'_j)^2 - \frac{1}{n} \sum_{i=1}^{n} g(\boldsymbol{x}_i) + \frac{\lambda}{2} R(g)^2 \right], \tag{9}$$

where $\mathcal{G}$ is some function space (i.e., a statistical model) and $R(\cdot)$ is some regularization functional.

3.1 Non-Parametric Convergence Analysis

First, we elucidate the non-parametric convergence rate of the proposed PE estimators. Here, we practically regard the function space $\mathcal{G}$ as an infinite-dimensional *reproducing kernel Hilbert space* (RKHS; Aronszajn, 1950) such as the Gaussian kernel space, and $R(\cdot)$ as the associated RKHS norm.



(a) $P' = N(0,1)$: P and P' are the same.

(b) $P' = N(0,0.6)$: P' has smaller standard deviation than P.

(c) $P' = N(0,2)$: P' has larger standard deviation than P.

(d) $P' = N(0.5,1)$: P and P' have different means.

(e) $P' = 0.95N(0,1) + 0.05N(3,1)$: P' contains an additional component to P.

Figure 1: Illustrative examples of density-ratio approximation by RuLSIF. From left to right: true densities ($P = N(0,1)$), true density-ratios, and their estimates for $\alpha = 0$, 0.5, and 0.95.

3.1.1 Theoretical Results

Let us represent the complexity of the function space $\mathcal{G}$ by γ $(0 < \gamma < 2)$; the larger γ is, the more complex the function class $\mathcal{G}$ is (see Appendix A for its precise definition). We analyze the convergence rate of our PE divergence estimators as $\bar{n} := \min(n, n')$ tends to infinity for $\lambda = \lambda_{\bar{n}}$ under

$$\lambda_{\bar{n}} \to o(1) \quad \text{and} \quad \lambda_{\bar{n}}^{-1} = o(\bar{n}^{2/(2+\gamma)}).$$

The first condition means that $\lambda_{\bar{n}}$ tends to zero, but the second condition means that its shrinking speed should not be too fast.

Under several technical assumptions detailed in Appendix A, we have the following asymptotic convergence results for the two PE divergence estimators $\widehat{\mathrm{PE}}_\alpha$ (7) and $\widetilde{\mathrm{PE}}_\alpha$ (8):

$$\widehat{\mathrm{PE}}_\alpha - \mathrm{PE}_\alpha = \mathcal{O}_p(\bar{n}^{-1/2} c \|r_\alpha\|_\infty + \lambda_{\bar{n}} \max(1, R(r_\alpha)^2)), \tag{10}$$

and

$$\begin{aligned}\widetilde{\mathrm{PE}}_\alpha - \mathrm{PE}_\alpha = \mathcal{O}_p\Big(&\lambda_{\bar{n}}^{1/2} \|r_\alpha\|_\infty^{1/2} \max\{1, R(r_\alpha)\} \\ &+ \lambda_{\bar{n}} \max\{1, \|r_\alpha\|_\infty^{(1-\gamma/2)/2}, R(r_\alpha)\|r_\alpha\|_\infty^{(1-\gamma/2)/2}, R(r_\alpha)\}\Big),\end{aligned} \tag{11}$$

where $\mathcal{O}_p$ denotes the asymptotic order in probability,

$$c := (1+\alpha)\sqrt{\mathbb{V}_{p(\boldsymbol{x})}[r_\alpha(\boldsymbol{x})]} + (1-\alpha)\sqrt{\mathbb{V}_{p'(\boldsymbol{x})}[r_\alpha(\boldsymbol{x})]}, \tag{12}$$

and $\mathbb{V}_{p(\boldsymbol{x})}[f(\boldsymbol{x})]$ denotes the variance of $f(\boldsymbol{x})$ under $p(\boldsymbol{x})$:

$$\mathbb{V}_{p(\boldsymbol{x})}[f(\boldsymbol{x})] = \int \left(f(\boldsymbol{x}) - \int f(\boldsymbol{x}) p(\boldsymbol{x}) \mathrm{d}\boldsymbol{x} \right)^2 p(\boldsymbol{x}) \mathrm{d}\boldsymbol{x}.$$

3.1.2 Interpretation

In both Eq.(10) and Eq.(11), the coefficients of the leading terms (i.e., the first terms) of the asymptotic convergence rates become smaller as $\|r_\alpha\|_\infty$ gets smaller. Since

$$\|r_\alpha\|_\infty = \left\| \big(\alpha + (1-\alpha)/r(\boldsymbol{x})\big)^{-1} \right\|_\infty < \tfrac{1}{\alpha} \quad \text{for } \alpha > 0,$$

larger α would be more preferable in terms of the asymptotic approximation error. Note that when $\alpha = 0$, $\|r_\alpha\|_\infty$ can tend to infinity even under a simple setting that the ratio of two Gaussian functions is considered (Cortes et al., 2010, see also the numerical examples in Section 2.4 of this paper). Thus, our proposed approach of estimating the α-relative PE divergence (with $\alpha > 0$) would be more advantageous than the naive approach of estimating the plain PE divergence (which corresponds to $\alpha = 0$) in terms of the non-parametric convergence rate.

The above results also show that $\widehat{\mathrm{PE}}_\alpha$ and $\widetilde{\mathrm{PE}}_\alpha$ have different asymptotic convergence rates. The leading term in Eq.(10) is of order $\bar{n}^{-1/2}$, while the leading term in Eq.(11) is of order $\lambda_{\bar{n}}^{1/2}$, which is slightly slower (depending on the complexity γ) than $\bar{n}^{-1/2}$. Thus, $\widehat{\mathrm{PE}}_\alpha$ would be more accurate than $\widetilde{\mathrm{PE}}_\alpha$ in large sample cases. Furthermore, when $p(\boldsymbol{x}) = p'(\boldsymbol{x})$, $\mathbb{V}_{p(\boldsymbol{x})}[r_\alpha(\boldsymbol{x})] = 0$ holds and thus $c = 0$ holds (see Eq.(12)). Then the leading term in Eq.(10) vanishes and therefore $\widehat{\mathrm{PE}}_\alpha$ has the even faster convergence rate of order $\lambda_{\bar{n}}$, which is slightly slower (depending on the complexity γ) than $\bar{n}^{-1}$. Similarly, if α is close to 1, $r_\alpha(\boldsymbol{x}) \approx 1$ and thus $c \approx 0$ holds.

When $\bar{n}$ is not large enough to be able to neglect the terms of $o(\bar{n}^{-1/2})$, the terms of $O(\lambda_{\bar{n}})$ matter. If $\|r_\alpha\|_\infty$ and $R(r_\alpha)$ are large (this can happen, e.g., when α is close to 0), the coefficient of the $O(\lambda_{\bar{n}})$-term in Eq.(10) can be larger than that in Eq.(11). Then $\widetilde{\mathrm{PE}}_\alpha$ would be more favorable than $\widehat{\mathrm{PE}}_\alpha$ in terms of the approximation accuracy.

3.1.3 Numerical Illustration

Let us numerically investigate the above interpretation using the same artificial dataset as Section 2.4.

Figure 2 shows the mean and standard deviation of $\widehat{\mathrm{PE}}_\alpha$ and $\widetilde{\mathrm{PE}}_\alpha$ over 100 runs for $\alpha = 0$, 0.5, and 0.95, as functions of n ($= n'$ in this experiment). The true PE_α (which was numerically computed) is also plotted in the graphs. The graphs show that both the estimators $\widehat{\mathrm{PE}}_\alpha$ and $\widetilde{\mathrm{PE}}_\alpha$ approach the true PE_α as the number of samples increases, and the approximation error tends to be smaller if α is larger.

When α is large, $\widehat{\mathrm{PE}}_\alpha$ tends to perform slightly better than $\widetilde{\mathrm{PE}}_\alpha$. On the other hand, when α is small and the number of samples is small, $\widetilde{\mathrm{PE}}_\alpha$ slightly compares favorably with $\widehat{\mathrm{PE}}_\alpha$. Overall, these numerical results well agree with our theory.

3.2 Parametric Variance Analysis

Next, we analyze the asymptotic variance of the PE divergence estimator $\widehat{\mathrm{PE}}_\alpha$ (7) under a parametric setup.

3.2.1 Theoretical Results

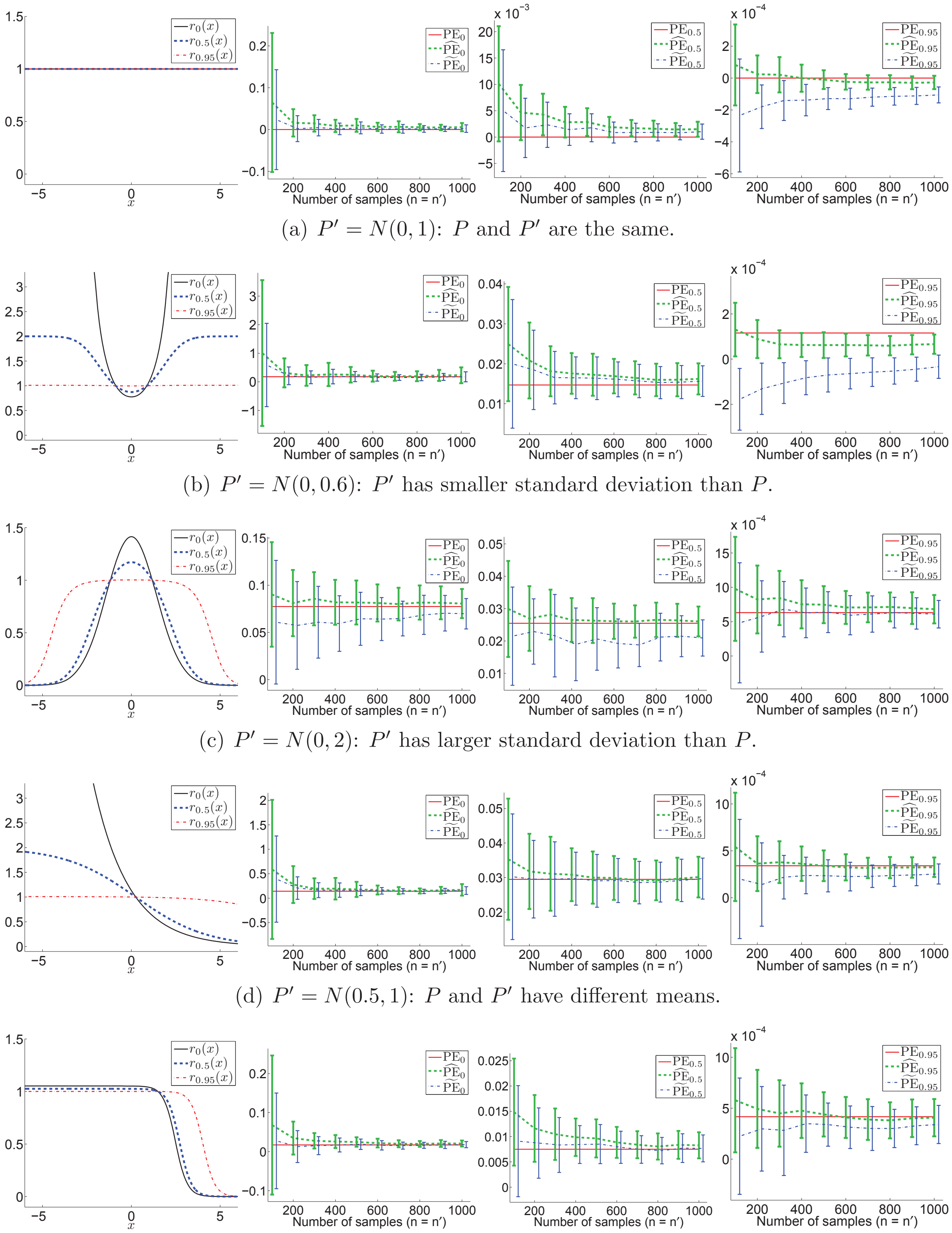
As the function space $\mathcal{G}$ in Eq.(9), we consider the following parametric model:

$$\mathcal{G} = \{g(\boldsymbol{x};\boldsymbol{\theta}) \mid \boldsymbol{\theta} \in \Theta \subset \mathbb{R}^b\},$$

where b is a finite number. Here we assume that the above parametric model is *correctly specified*, i.e., it includes the true relative density-ratio function $r_\alpha(\boldsymbol{x})$: there exists $\boldsymbol{\theta}^*$ such that

$$g(\boldsymbol{x};\boldsymbol{\theta}^*) = r_\alpha(\boldsymbol{x}).$$

Here, we use RuLSIF without regularization, i.e., $\lambda = 0$ in Eq.(9).



(a) $P' = N(0,1)$: P and P' are the same.

(b) $P' = N(0,0.6)$: P' has smaller standard deviation than P.

(c) $P' = N(0,2)$: P' has larger standard deviation than P.

(d) $P' = N(0.5,1)$: P and P' have different means.

(e) $P' = 0.95N(0,1) + 0.05N(3,1)$: P' contains an additional component to P.

Figure 2: Illustrative examples of divergence estimation by RuLSIF. From left to right: true density-ratios for $\alpha = 0$, 0.5, and 0.95 ($P = N(0,1)$), and estimation error of PE divergence for $\alpha = 0$, 0.5, and 0.95.

Let us denote the variance of $\widehat{\mathrm{PE}}_\alpha$ (7) by $\mathbb{V}[\widehat{\mathrm{PE}}_\alpha]$, where randomness comes from the draw of samples $\{\boldsymbol{x}_i\}_{i=1}^n$ and $\{\boldsymbol{x}'_j\}_{j=1}^{n'}$. Then, under a standard regularity condition for the asymptotic normality (see Section 3 of van der Vaart, 2000), $\mathbb{V}[\widehat{\mathrm{PE}}_\alpha]$ can be expressed and upper-bounded as

$$\mathbb{V}[\widehat{\mathrm{PE}}_\alpha] = \frac{1}{n}\mathbb{V}_{p(\boldsymbol{x})}\left[r_\alpha - \frac{\alpha r_\alpha(\boldsymbol{x})^2}{2}\right] + \frac{1}{n'}\mathbb{V}_{p'(\boldsymbol{x})}\left[\frac{(1-\alpha)r_\alpha(\boldsymbol{x})^2}{2}\right] + o\left(\frac{1}{n}, \frac{1}{n'}\right) \tag{13}$$

$$\leq \frac{\|r_\alpha\|_\infty^2}{n} + \frac{\alpha^2\|r_\alpha\|_\infty^4}{4n} + \frac{(1-\alpha)^2\|r_\alpha\|_\infty^4}{4n'} + o\left(\frac{1}{n}, \frac{1}{n'}\right). \tag{14}$$

Let us denote the variance of $\widetilde{\mathrm{PE}}_\alpha$ by $\mathbb{V}[\widetilde{\mathrm{PE}}_\alpha]$. Then, under a standard regularity condition for the asymptotic normality (see Section 3 of van der Vaart, 2000), the variance of $\widetilde{\mathrm{PE}}_\alpha$ is asymptotically expressed as

$$\begin{aligned}\mathbb{V}[\widetilde{\mathrm{PE}}_\alpha] = {} & \frac{1}{n}\mathbb{V}_{p(\boldsymbol{x})}\left[\frac{r_\alpha + (1-\alpha r_\alpha)\mathbb{E}_{p(\boldsymbol{x})}[\nabla g]^\top \boldsymbol{U}_\alpha^{-1}\nabla g}{2}\right] \\ & + \frac{1}{n'}\mathbb{V}_{p'(\boldsymbol{x})}\left[\frac{(1-\alpha)r_\alpha\mathbb{E}_{p(\boldsymbol{x})}[\nabla g]^\top \boldsymbol{U}_\alpha^{-1}\nabla g}{2}\right] + o\left(\frac{1}{n}, \frac{1}{n'}\right),\end{aligned} \tag{15}$$

where ∇g is the gradient vector of g with respect to $\boldsymbol{\theta}$ at $\boldsymbol{\theta} = \boldsymbol{\theta}^*$, i.e.,

$$(\nabla g(\boldsymbol{x};\boldsymbol{\theta}^*))_j = \frac{\partial g(\boldsymbol{x};\boldsymbol{\theta}^*)}{\partial \theta_j}.$$

The matrix $\boldsymbol{U}_\alpha$ is defined by

$$\boldsymbol{U}_\alpha = \alpha\mathbb{E}_{p(\boldsymbol{x})}[\nabla g \nabla g^\top] + (1-\alpha)\mathbb{E}_{p'(\boldsymbol{x})}[\nabla g \nabla g^\top].$$

3.2.2 Interpretation

Eq.(13) shows that, up to $O\left(\frac{1}{n}, \frac{1}{n'}\right)$, the variance of $\widehat{\mathrm{PE}}_\alpha$ depends only on the true relative density-ratio $r_\alpha(\boldsymbol{x})$, not on the estimator of $r_\alpha(\boldsymbol{x})$. This means that the model complexity does not affect the asymptotic variance. Therefore, *overfitting* would hardly occur in the estimation of the relative PE divergence even when complex models are used. We note that the above superior property is applicable only to relative PE divergence estimation, not to relative density-ratio estimation. This implies that overfitting occurs in relative density-ratio estimation, but the approximation error cancels out in relative PE divergence estimation.

On the other hand, Eq.(15) shows that the variance of $\widetilde{\mathrm{PE}}_\alpha$ is affected by the model $\mathcal{G}$, since the factor $\mathbb{E}_{p(\boldsymbol{x})}[\nabla g]^\top \boldsymbol{U}_\alpha^{-1}\nabla g$ depends on the model complexity in general. When the equality

$$\mathbb{E}_{p(\boldsymbol{x})}[\nabla g]^\top \boldsymbol{U}_\alpha^{-1}\nabla g(\boldsymbol{x};\boldsymbol{\theta}^*) = r_\alpha(\boldsymbol{x})$$

holds, the variances of $\widetilde{\mathrm{PE}}_\alpha$ and $\widehat{\mathrm{PE}}_\alpha$ are asymptotically the same. However, in general, the use of $\widehat{\mathrm{PE}}_\alpha$ would be more recommended.

Eq.(14) shows that the variance $\mathbb{V}[\widehat{\mathrm{PE}}_\alpha]$ can be upper-bounded by the quantity depending on $\|r_\alpha\|_\infty$, which is monotonically lowered if $\|r_\alpha\|_\infty$ is reduced. Since $\|r_\alpha\|_\infty$ monotonically decreases as α increases, our proposed approach of estimating the α-relative PE divergence (with $\alpha > 0$) would be more advantageous than the naive approach of estimating the plain PE divergence (which corresponds to $\alpha = 0$) in terms of the parametric asymptotic variance.

3.2.3 Numerical Illustration

Here, we show some numerical results for illustrating the above theoretical results using the one-dimensional datasets (b) and (c) in Section 2.4. Let us define the parametric model as

$$\mathcal{G}_k = \left\{ g(x;\boldsymbol{\theta}) = \frac{r(x;\boldsymbol{\theta})}{\alpha r(x;\boldsymbol{\theta}) + 1 - \alpha} \;\middle|\; r(x;\boldsymbol{\theta}) = \exp\left(\sum_{\ell=0}^{k} \theta_\ell x^\ell \right), \boldsymbol{\theta} \in \mathbb{R}^{k+1} \right\}. \tag{16}$$

The dimension of the model $\mathcal{G}_k$ is equal to $k+1$. The α-relative density-ratio $r_\alpha(x)$ can be expressed using the ordinary density-ratio $r(x) = p(x)/p'(x)$ as

$$r_\alpha(x) = \frac{r(x)}{\alpha r(x) + 1 - \alpha}.$$

Thus, when $k > 1$, the above model $\mathcal{G}_k$ includes the true relative density-ratio $r_\alpha(x)$ of the datasets (b) and (c). We test RuLSIF with $\alpha = 0.2$ and 0.8 for the model (16) with degree $k = 1, 2, \ldots, 8$. The parameter $\boldsymbol{\theta}$ is learned so that Eq.(9) is minimized by a quasi-Newton method.

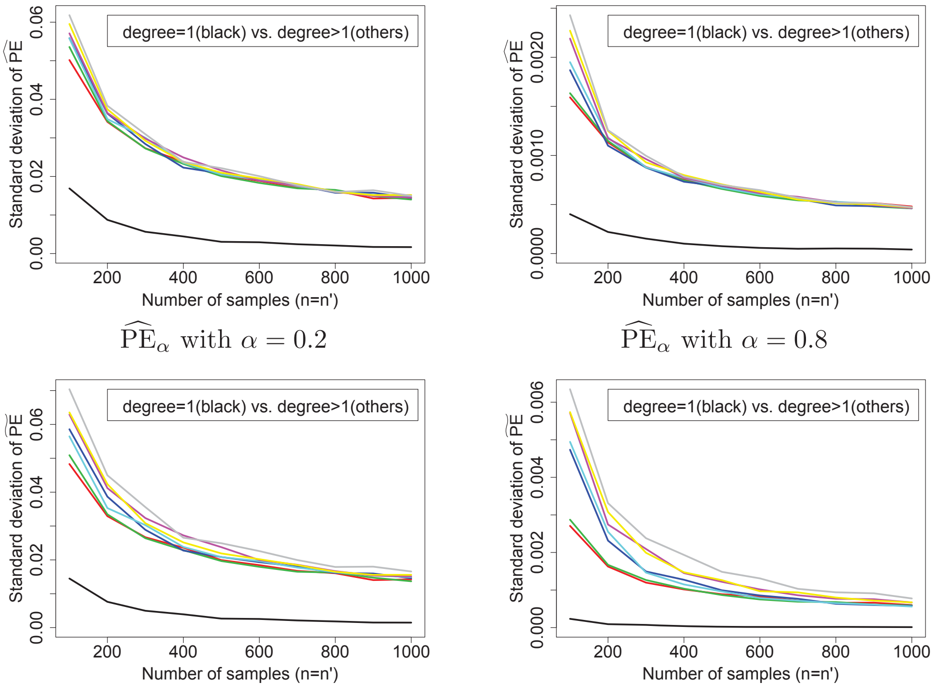
The standard deviations of $\widehat{\mathrm{PE}}_\alpha$ and $\widetilde{\mathrm{PE}}_\alpha$ for the datasets (b) and (c) are depicted in Figure 3 and Figure 4, respectively. The graphs show that the degree of models does not significantly affect the standard deviation of $\widehat{\mathrm{PE}}_\alpha$ (i.e., no overfitting), as long as the model includes the true relative density-ratio (i.e., $k > 1$). On the other hand, bigger models tend to produce larger standard deviations in $\widetilde{\mathrm{PE}}_\alpha$. Thus, the standard deviation of $\widetilde{\mathrm{PE}}_\alpha$ more strongly depends on the model complexity.

4 Experiments

In this section, we experimentally evaluate the performance of the proposed method in two-sample homogeneity test, outlier detection, and transfer learning tasks.

4.1 Two-Sample Homogeneity Test

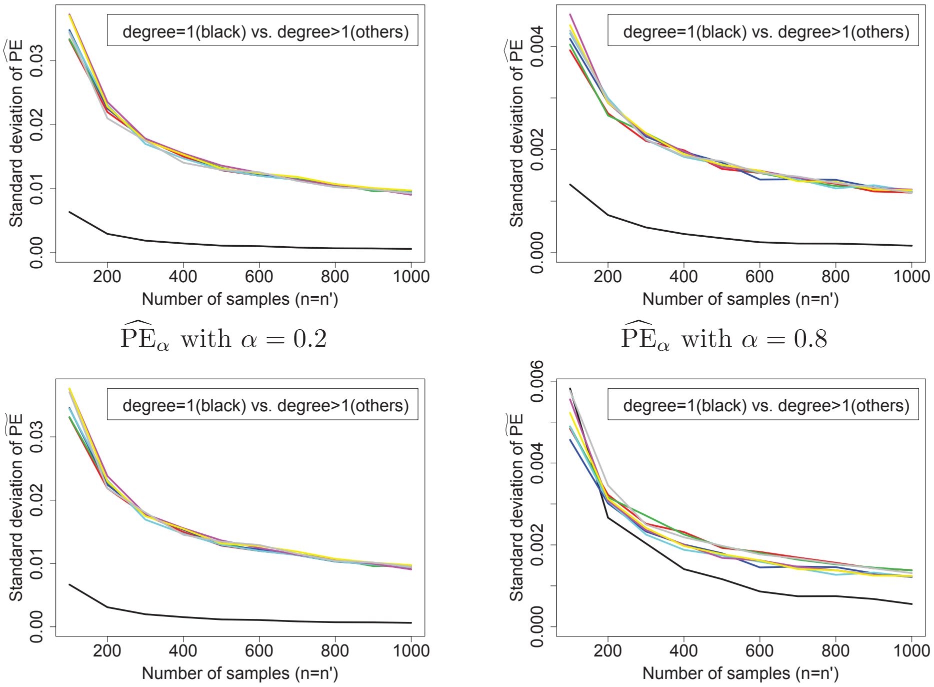
First, we apply the proposed divergence estimator to two-sample homogeneity test.



$\widehat{\mathrm{PE}}_\alpha$ with $\alpha = 0.2$ $\widehat{\mathrm{PE}}_\alpha$ with $\alpha = 0.8$

$\widetilde{\mathrm{PE}}_\alpha$ with $\alpha = 0.2$ $\widetilde{\mathrm{PE}}_\alpha$ with $\alpha = 0.8$

Figure 3: Standard deviations of PE estimators for dataset (b) (i.e., $P = N(0,1)$ and $P' = N(0,0.6)$) as functions of the sample size $n = n'$.



$\widehat{\mathrm{PE}}_\alpha$ with $\alpha = 0.2$ $\widehat{\mathrm{PE}}_\alpha$ with $\alpha = 0.8$

$\widetilde{\mathrm{PE}}_\alpha$ with $\alpha = 0.2$ $\widetilde{\mathrm{PE}}_\alpha$ with $\alpha = 0.8$

Figure 4: Standard deviations of PE estimators for dataset (c) (i.e., $P = N(0,1)$ and $P' = N(0,2)$) as functions of the sample size $n = n'$.

4.1.1 Divergence-Based Two-Sample Homogeneity Test

Given two sets of samples $\mathcal{X} = \{\boldsymbol{x}_i\}_{i=1}^{n} \overset{\text{i.i.d.}}{\sim} P$ and $\mathcal{X}' = \{\boldsymbol{x}'_j\}_{j=1}^{n'} \overset{\text{i.i.d.}}{\sim} P'$, the goal of the two-sample homogeneity test is to test the *null hypothesis* that the probability distributions P and P' are the same against its complementary alternative (i.e., the distributions are different).

By using an estimator $\widehat{\mathrm{Div}}$ of some divergence between the two distributions P and P', homogeneity of two distributions can be tested based on the *permutation test* procedure (Efron and Tibshirani, 1993) as follows:

- Obtain a divergence estimate $\widehat{\mathrm{Div}}$ using the original datasets $\mathcal{X}$ and $\mathcal{X}'$.
- Randomly permute the $|\mathcal{X} \cup \mathcal{X}'|$ samples, and assign the first $|\mathcal{X}|$ samples to a set $\widetilde{\mathcal{X}}$ and the remaining $|\mathcal{X}'|$ samples to another set $\widetilde{\mathcal{X}}'$.
- Obtain a divergence estimate $\widetilde{\mathrm{Div}}$ using the randomly shuffled datasets $\widetilde{\mathcal{X}}$ and $\widetilde{\mathcal{X}}'$ (note that, since $\widetilde{\mathcal{X}}$ and $\widetilde{\mathcal{X}}'$ can be regarded as being drawn from the same distribution, $\widetilde{\mathrm{Div}}$ tends to be close to zero).
- Repeat this random shuffling procedure many times, and construct the empirical distribution of $\widetilde{\mathrm{Div}}$ under the null hypothesis that the two distributions are the same.
- Approximate the p-value by evaluating the relative ranking of the original $\widehat{\mathrm{Div}}$ in the distribution of $\widetilde{\mathrm{Div}}$.

When an asymmetric divergence such as the KL divergence (Kullback and Leibler, 1951) or the PE divergence (Pearson, 1900) is adopted for two-sample homogeneity test, the test results depend on the choice of *directions*: a divergence from P to P' or from P' to P. (Sugiyama et al., 2011) proposed to choose the direction that gives a smaller p-value—it was experimentally shown that, when the uLSIF-based PE divergence estimator is used for the two-sample homogeneity test (which is called the *least-squares two-sample homogeneity test*; LSTT), the heuristic of choosing the direction with a smaller p-value contributes to reducing the *type-II error* (the probability of accepting incorrect null-hypotheses, i.e., two distributions are judged to be the same when they are actually different), while the increase of the *type-I error* (the probability of rejecting correct null-hypotheses, i.e., two distributions are judged to be different when they are actually the same) is kept moderate.

Below, we refer to LSTT with $p(\boldsymbol{x})/p'(\boldsymbol{x})$ as the *plain LSTT*, LSTT with $p'(\boldsymbol{x})/p(\boldsymbol{x})$ as the *reciprocal LSTT*, and LSTT with heuristically choosing the one with a smaller p-value as the *adaptive LSTT*.

4.1.2 Artificial Datasets

We illustrate how the proposed method behaves in two-sample homogeneity test scenarios using the artificial datasets (a)–(d) described in Section 2.4. We test the plain LSTT, reciprocal LSTT, and adaptive LSTT for $\alpha = 0$, 0.5, and 0.95, with significance level 5%.

The experimental results are shown in Figure 5. For the dataset (a) where $P = P'$ (i.e., the null hypothesis is correct), the plain LSTT and reciprocal LSTT correctly accept the null hypothesis with probability approximately 95%. This means that the type-I error is properly controlled in these methods. On the other hand, the adaptive LSTT tends to give slightly lower acceptance rates than 95% for this toy dataset, but the adaptive LSTT with $\alpha = 0.5$ still works reasonably well. This implies that the heuristic of choosing the method with a smaller p-value does not have critical influence on the type-I error.

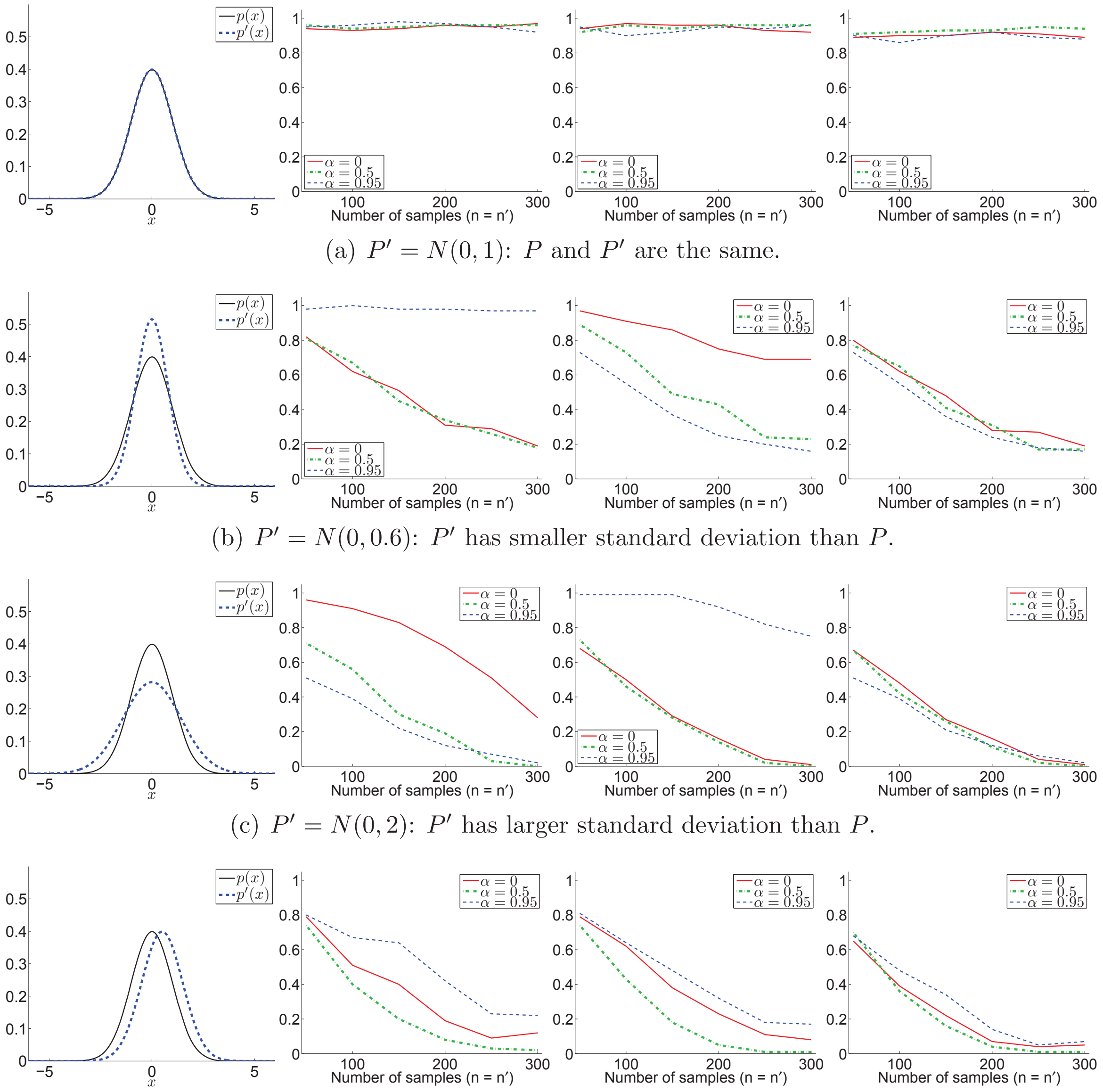
In the datasets (b), (c), and (d), P is different from P' (i.e., the null hypothesis is not correct), and thus we want to reduce the acceptance rate of the incorrect null-hypothesis as much as possible. In the plain setup for the dataset (b) and the reciprocal setup for the dataset (c), the true density-ratio functions with $\alpha = 0$ diverge to infinity, and thus larger α makes the density-ratio approximation more reliable. However, $\alpha = 0.95$ does not work well because it produces an overly-smoothed density-ratio function and thus it is hard to be distinguished from the completely constant density-ratio function (which corresponds to $P = P'$). On the other hand, in the reciprocal setup for the dataset (b) and the plain setup for the dataset (c), small α performs poorly since density-ratio functions with large α can be more accurately approximated than those with small α (see Figure 1). In the adaptive setup, large α tends to perform slightly better than small α for the datasets (b) and (c).

In the dataset (d), the true density-ratio function with $\alpha = 0$ diverges to infinity for both the plain and reciprocal setups. In this case, middle α performs the best, which well balances the trade-off between high distinguishability from the completely constant density-ratio function (which corresponds to $P = P'$) and easy approximability. The same tendency that middle α works well can also be mildly observed in the adaptive LSTT for the dataset (d).

Overall, if the plain LSTT (or the reciprocal LSTT) is used, small α (or large α) sometimes works excellently. However, it performs poorly in other cases and thus the performance is unstable depending on the true distributions. The plain LSTT (or the reciprocal LSTT) with middle α tends to perform reasonably well for all datasets. On the other hand, the adaptive LSTT was shown to nicely overcome the above instability problem when α is small or large. However, when α is set to be a middle value, the plain LSTT and the reciprocal LSTT both give similar results and thus the adaptive LSTT provides only a small amount of improvement.

Our empirical finding is that, if we have prior knowledge that one distribution has a wider support than the other distribution, assigning the distribution with a wider support to P' and setting α to be a large value seem to work well. If there is no knowledge on the true distributions or two distributions have less overlapped supports, using middle α in the adaptive setup seems to be a reasonable choice.

We will systematically investigate this issue using more complex datasets below.



(a) $P' = N(0,1)$: P and P' are the same.

(b) $P' = N(0,0.6)$: P' has smaller standard deviation than P.

(c) $P' = N(0,2)$: P' has larger standard deviation than P.

(d) $P' = N(0.5,1)$: P and P' have different means.

Figure 5: Illustrative examples of two-sample homogeneity test based on relative divergence estimation. From left to right: true densities ($P = N(0,1)$), the acceptance rate of the null hypothesis under the significance level 5% by plain LSTT, reciprocal LSTT, and adaptive LSTT.

4.1.3 Benchmark Datasets

Here, we apply the proposed two-sample homogeneity test to the binary classification datasets taken from the *IDA repository* (Rätsch et al., 2001).

We test the adaptive LSTT with the RuLSIF-based PE divergence estimator for $\alpha = 0$, 0.5, and 0.95; we also test the *maximum mean discrepancy* (MMD; Borgwardt et al., 2006), which is a kernel-based two-sample homogeneity test method. The performance of MMD depends on the choice of the Gaussian kernel width. Here, we adopt a version proposed by (Sriperumbudur et al., 2009), which automatically optimizes the Gaussian kernel width. The p-values of MMD are computed in the same way as LSTT based on the permutation test procedure.

First, we investigate the rate of accepting the null hypothesis when the null hypothesis is correct (i.e., the two distributions are the same). We split all the positive training samples into two sets and perform two-sample homogeneity test for the two sets of samples. The experimental results are summarized in Table 1, showing that the adaptive LSTT with $\alpha = 0.5$ compares favorably with those with $\alpha = 0$ and 1 and MMD in terms of the type-I error.

Next, we consider the situation where the null hypothesis is not correct (i.e., the two distributions are different). The numerator samples are generated in the same way as above, but a half of denominator samples are replaced with negative training samples. Thus, while the numerator sample set contains only positive training samples, the denominator sample set includes both positive and negative training samples. The experimental results are summarized in Table 2, showing that the adaptive LSTT with $\alpha = 0.5$ again compares favorably with those with $\alpha = 0$ and 1. Furthermore, LSTT with $\alpha = 0.5$ tends to outperform MMD in terms of the type-II error.

Overall, LSTT with $\alpha = 0.5$ is shown to be a useful method for two-sample homogeneity test.

4.2 Inlier-Based Outlier Detection

Next, we apply the proposed method to outlier detection.

4.2.1 Density-Ratio Approach to Inlier-Based Outlier Detection

Let us consider an outlier detection problem of finding irregular samples in a dataset (called an "evaluation dataset") based on another dataset (called a "model dataset") that only contains regular samples. Defining the density ratio over the two sets of samples, we can see that the density-ratio values for regular samples are close to one, while those for outliers tend to be significantly deviated from one. Thus, density-ratio values could be used as an index of the degree of outlyingness (Smola et al., 2009; Hido et al., 2011).

Since the evaluation dataset usually has a wider support than the model dataset, we regard the evaluation dataset as samples corresponding to the denominator density $p'(\boldsymbol{x})$, and the model dataset as samples corresponding to the numerator density $p(\boldsymbol{x})$.

Table 1: Experimental results of two-sample homogeneity test for the IDA datasets. The mean (and standard deviation in the bracket) rate of accepting the null hypothesis (i.e., $P = P'$) under the significance level 5% is reported. The two sets of samples are both taken from the positive training set (i.e., the null hypothesis is correct). Methods having the mean acceptance rate 0.95 according to the one-sample t-test at the significance level 5% are specified by bold face.

Datasets	d	$n = n'$	MMD	LSTT ($\alpha = 0.0$)	LSTT ($\alpha = 0.5$)	LSTT ($\alpha = 0.95$)
banana	2	100	**0.96(0.20)**	**0.93(0.26)**	**0.92(0.27)**	**0.92(0.27)**
thyroid	5	19	**0.96(0.20)**	**0.95(0.22)**	**0.95(0.22)**	0.88 (0.33)
titanic	5	21	**0.94(0.24)**	0.86 (0.35)	**0.92(0.27)**	**0.89(0.31)**
diabetes	8	85	**0.96(0.20)**	0.87 (0.34)	**0.91(0.29)**	0.82 (0.39)
breast-cancer	9	29	0.98 (0.14)	**0.91(0.29)**	**0.94(0.24)**	**0.92(0.27)**
flare-solar	9	100	**0.93(0.26)**	**0.91(0.29)**	**0.95(0.22)**	**0.93(0.26)**
heart	13	38	1.00 (0.00)	0.85 (0.36)	**0.91(0.29)**	**0.93(0.26)**
german	20	100	0.99 (0.10)	**0.91(0.29)**	**0.92(0.27)**	**0.89(0.31)**
ringnorm	20	100	**0.97(0.17)**	**0.93(0.26)**	**0.91(0.29)**	0.85 (0.36)
waveform	21	66	0.98 (0.14)	**0.92(0.27)**	**0.93(0.26)**	0.88 (0.33)

Table 2: Experimental results of two-sample homogeneity test for the IDA datasets. The mean (and standard deviation in the bracket) rate of accepting the null hypothesis (i.e., $P = P'$) under the significance level 5% is reported. The set of samples corresponding to the numerator of the density ratio is taken from the positive training set and the set of samples corresponding to the denominator of the density ratio is taken from the positive training set and the negative training set (i.e., the null hypothesis is not correct). The best method having the lowest mean acceptance rate and comparable methods according to the *two-sample t-test* at the significance level 5% are specified by bold face.

Datasets	d	$n = n'$	MMD	LSTT ($\alpha = 0.0$)	LSTT ($\alpha = 0.5$)	LSTT ($\alpha = 0.95$)
banana	2	100	0.52 (0.50)	**0.10(0.30)**	**0.02(0.14)**	**0.17(0.38)**
thyroid	5	19	**0.52(0.50)**	0.81 (0.39)	**0.65(0.48)**	0.80 (0.40)
titanic	5	21	**0.87(0.34)**	**0.86(0.35)**	**0.87(0.34)**	**0.88(0.33)**
diabetes	8	85	**0.31(0.46)**	**0.42(0.50)**	0.47 (0.50)	0.57 (0.50)
breast-cancer	9	29	0.87 (0.34)	**0.75(0.44)**	0.80 (0.40)	0.79 (0.41)
flare-solar	9	100	**0.51(0.50)**	0.81 (0.39)	**0.55(0.50)**	**0.66(0.48)**
heart	13	38	0.53 (0.50)	**0.28(0.45)**	**0.40(0.49)**	0.62 (0.49)
german	20	100	0.56 (0.50)	0.55 (0.50)	**0.44(0.50)**	0.68 (0.47)
ringnorm	20	100	**0.00(0.00)**	**0.00(0.00)**	**0.00(0.00)**	**0.02(0.14)**
waveform	21	66	**0.00(0.00)**	**0.00(0.00)**	**0.02(0.14)**	**0.00(0.00)**

Table 3: Mean AUC score (and the standard deviation in the bracket) over 1000 trials for the artificial outlier-detection dataset. The best method in terms of the mean AUC score and comparable methods according to the *two-sample t-test* at the significance level 5% are specified by bold face.

Input dimensionality d	RuLSIF ($\alpha = 0$)	RuLSIF ($\alpha = 0.5$)	RuLSIF ($\alpha = 0.95$)
1	**.933(.089)**	**.926(.100)**	.896 (.124)
5	**.882(.099)**	**.891(.091)**	**.894(.086)**
10	.842 (.107)	**.850(.103)**	**.859(.092)**

Then, outliers tend to have smaller density-ratio values (i.e., close to zero). As such, density-ratio approximators can be used for outlier detection.

When evaluating the performance of outlier detection methods, it is important to take into account both the *detection rate* (i.e., the amount of true outliers an outlier detection algorithm can find) and the *detection accuracy* (i.e., the amount of true inliers an outlier detection algorithm misjudges as outliers). Since there is a trade-off between the detection rate and the detection accuracy, we adopt the *area under the ROC curve* (AUC) as our error metric (Bradley, 1997).

4.2.2 Artificial Datasets

First, we illustrate how the proposed method behaves in outlier detection scenarios using artificial datasets.

Let

$$\begin{aligned} P &= N(0, \boldsymbol{I}_d), \\ P' &= 0.95N(0, \boldsymbol{I}_d) + 0.05N(3d^{-1/2}\mathbf{1}_d, \boldsymbol{I}_d), \end{aligned}$$

where d is the dimensionality of $\boldsymbol{x}$ and $\mathbf{1}_d$ is the d-dimensional vector with all one. Note that this setup is the same as the dataset (e) described in Section 2.4 when $d = 1$. Here, the samples drawn from $N(0, \boldsymbol{I}_d)$ are regarded as inliers, while the samples drawn from $N(d^{-1/2}\mathbf{1}_d, \boldsymbol{I}_d)$ are regarded as outliers. We use $n = n' = 100$ samples.

Table 3 describes the AUC values for input dimensionality $d = 1$, 5, and 10 for RuLSIF with $\alpha = 0$, 0.5, and 0.95. This shows that, as the input dimensionality d increases, the AUC values overall get smaller. Thus, outlier detection becomes more challenging in high-dimensional cases.

The result also shows that RuLSIF with small α tends to work well when the input dimensionality is low, and RuLSIF with large α works better as the input dimensionality increases. This tendency can be interpreted as follows: If α is small, the density-ratio function tends to have sharp ‘hollow’ for outlier points (see the leftmost graph in Figure 2(e)). Thus, as long as the true density-ratio function can be accurately estimated, small α would be preferable in outlier detection. When the data dimensionality is low, density-ratio approximation is rather easy and thus small α tends to perform well. However, as the data dimensionality increases, density-ratio approximation gets harder, and thus large α which produces a smoother density-ratio function is more favorable since such a smoother function can be more easily approximated than a 'bumpy' one produced by small α.

4.2.3 Real-World Datasets

Next, we evaluate the proposed outlier detection method using various real-world datasets:

IDA repository: The *IDA repository* (Rätsch et al., 2001) contains various binary classification tasks. Each dataset consists of positive/negative and training/test samples. We use positive training samples as inliers in the "model" set. In the "evaluation" set, we use at most 100 positive test samples as inliers and the first 5% of negative test samples as outliers. Thus, the positive samples are treated as inliers and the negative samples are treated as outliers.

Speech dataset: An in-house speech dataset, which contains short utterance samples recorded from 2 male subjects speaking in French with sampling rate 44.1kHz. From each utterance sample, we extracted a 50-dimensional *line spectral frequencies* vector (Kain and Macon, 1998). We randomly take 200 samples from one class and assign them to the model dataset. Then we randomly take 200 samples from the same class and 10 samples from the other class.

20 Newsgroup dataset: The *20-Newsgroups* dataset[1] contains 20000 newsgroup documents, which contains the following 4 top-level categories: 'comp', 'rec', 'sci', and 'talk'. Each document is expressed by a 100-dimensional bag-of-words vector of term-frequencies. We randomly take 200 samples from the 'comp' class and assign them to the model dataset. Then we randomly take 200 samples from the same class and 10 samples from one of the other classes for the evaluation dataset.

The USPS hand-written digit dataset: The *USPS* hand-written digit dataset[2] contains 9298 digit images. Each image consists of 256 ($= 16 \times 16$) pixels and each pixel takes an integer value between 0 and 255 as the intensity level. We regard samples in one class as inliers and samples in other classes as outliers. We randomly take 200 samples from the inlier class and assign them to the model dataset. Then we randomly take 200 samples from the same inlier class and 10 samples from one of the other classes for the evaluation dataset.

We compare the AUC scores of RuLSIF with $\alpha = 0$, 0.5, and 0.95, and *one-class support vector machine (OSVM)* with the Gaussian kernel (Schölkopf et al., 2001). We used the *LIBSVM* implementation of OSVM (Chang and Lin, 2001). The Gaussian width is set to the median distance between samples, which has been shown to be a useful

[1] `http://people.csail.mit.edu/jrennie/20Newsgroups/`

[2] `http://www.gaussianprocess.org/gpml/data/`

Table 4: Experimental results of outlier detection for various for real-world datasets. Mean AUC score (and standard deviation in the bracket) over 100 trials is reported. The best method having the highest mean AUC score and comparable methods according to the *two-sample t-test* at the significance level 5% are specified by bold face. The datasets are sorted in the ascending order of the input dimensionality d.

Datasets	d	OSVM ($\nu = 0.05$)	OSVM ($\nu = 0.1$)	RuLSIF ($\alpha = 0$)	RuLSIF ($\alpha = 0.5$)	RuLSIF ($\alpha = 0.95$)
IDA:banana	2	**.668(.105)**	**.676(.120)**	.597 (.097)	.619 (.101)	.623 (.115)
IDA:thyroid	5	.760 (.148)	**.782(.165)**	**.804(.148)**	**.796(.178)**	.722 (.153)
IDA:titanic	5	**.757(.205)**	**.752(.191)**	**.750(.182)**	.701 (.184)	.712 (.185)
IDA:diabetes	8	**.636(.099)**	.610 (.090)	.594 (.105)	.575 (.105)	**.663(.112)**
IDA:b-cancer	9	**.741(.160)**	.691 (.147)	**.707(.148)**	**.737(.159)**	**.733(.160)**
IDA:f-solar	9	.594 (.087)	.590 (.083)	**.626(.102)**	**.612(.100)**	.584 (.114)
IDA:heart	13	.714 (.140)	.694 (.148)	**.748(.149)**	**.769(.134)**	.726 (.127)
IDA:german	20	**.612(.069)**	**.604(.084)**	**.605(.092)**	**.597(.101)**	**.605(.095)**
IDA:ringnorm	20	**.991(.012)**	**.993(.007)**	.944 (.091)	.971 (.062)	**.992(.010)**
IDA:waveform	21	.812 (.107)	.843 (.123)	**.879(.122)**	**.875(.117)**	**.885(.102)**
Speech	50	.788 (.068)	**.830(.060)**	.804 (.101)	**.821(.076)**	**.836(.083)**
20News ('rec')	100	.598 (.063)	.593 (.061)	.628 (.105)	.614 (.093)	**.767(.100)**
20News ('sci')	100	.592 (.069)	.589 (.071)	.620 (.094)	.609 (.087)	**.704(.093)**
20News ('talk')	100	.661 (.084)	.658 (.084)	.672 (.117)	.670 (.102)	**.823(.078)**
USPS (1 vs. 2)	256	.889 (.052)	**.926(.037)**	.848 (.081)	.878 (.088)	.898 (.051)
USPS (2 vs. 3)	256	.823 (.053)	.835 (.050)	.803 (.093)	.818 (.085)	**.879(.074)**
USPS (3 vs. 4)	256	.901 (.044)	.939 (.031)	.950 (.056)	.961 (.041)	**.984(.016)**
USPS (4 vs. 5)	256	.871 (.041)	.890 (.036)	.857 (.099)	.874 (.082)	**.941(.031)**
USPS (5 vs. 6)	256	.825 (.058)	.859 (.052)	.863 (.078)	.867 (.068)	**.901(.049)**
USPS (6 vs. 7)	256	.910 (.034)	.950 (.025)	.972 (.038)	.984 (.018)	**.994(.010)**
USPS (7 vs. 8)	256	.938 (.030)	.967 (.021)	.941 (.053)	.951 (.039)	**.980(.015)**
USPS (8 vs. 9)	256	.721 (.072)	.728 (.073)	.721 (.084)	.728 (.083)	**.761(.096)**
USPS (9 vs. 0)	256	.920 (.037)	.966 (.023)	.982 (.048)	.989 (.022)	**.994(.011)**

heuristic (Schölkopf et al., 2001). Since there is no systematic method to determine the tuning parameter ν in OSVM, we report the results for $\nu = 0.05$ and 0.1.

The mean and standard deviation of the AUC scores over 100 runs with random sample choice are summarized in Table 4, showing that RuLSIF overall compares favorably with OSVM. Among the RuLSIF methods, small α tends to perform well for low-dimensional datasets, and large α tends to work well for high-dimensional datasets. This tendency well agrees with that for the artificial datasets (see Section 4.2.2).

4.3 Transfer Learning

Finally, we apply the proposed method to transfer learning.

4.3.1 Transductive Transfer Learning by Importance Sampling

Let us consider a problem of *semi-supervised learning* (Chapelle et al., 2006) from labeled training samples $\{(\boldsymbol{x}_j^{\mathrm{tr}}, y_j^{\mathrm{tr}})\}_{j=1}^{n_{\mathrm{tr}}}$ and unlabeled test samples $\{\boldsymbol{x}_i^{\mathrm{te}}\}_{i=1}^{n_{\mathrm{te}}}$. The goal is to predict a test output value y^{te} for a test input point $\boldsymbol{x}^{\mathrm{te}}$. Here, we consider the setup where the labeled training samples $\{(\boldsymbol{x}_j^{\mathrm{tr}}, y_j^{\mathrm{tr}})\}_{j=1}^{n_{\mathrm{tr}}}$ are drawn i.i.d. from $p(y|\boldsymbol{x})p_{\mathrm{tr}}(\boldsymbol{x})$, while the unlabeled test samples $\{\boldsymbol{x}_i^{\mathrm{te}}\}_{i=1}^{n_{\mathrm{te}}}$ are drawn i.i.d. from $p_{\mathrm{te}}(\boldsymbol{x})$, which is generally different from $p_{\mathrm{tr}}(\boldsymbol{x})$; the (unknown) test sample $(\boldsymbol{x}^{\mathrm{te}}, y^{\mathrm{te}})$ follows $p(y|\boldsymbol{x})p_{\mathrm{te}}(\boldsymbol{x})$. This setup means that the conditional probability $p(y|\boldsymbol{x})$ is common to training and test samples, but the marginal densities $p_{\mathrm{tr}}(\boldsymbol{x})$ and $p_{\mathrm{te}}(\boldsymbol{x})$ are generally different for training and test input points. Such a problem is called *transductive transfer learning* (Pan and Yang, 2010), *domain adaptation* (Jiang and Zhai, 2007), or *covariate shift* (Shimodaira, 2000; Sugiyama and Kawanabe, 2012).

Let $\mathrm{loss}(y, \widehat{y})$ be a point-wise loss function that measures a discrepancy between y and $\widehat{y}$ (at input $\boldsymbol{x}$). Then the *generalization error* which we would like to ultimately minimize is defined as

$$\mathbb{E}_{p(y|\boldsymbol{x})p_{\mathrm{te}}(\boldsymbol{x})}\left[\mathrm{loss}(y, f(\boldsymbol{x}))\right],$$

where $f(\boldsymbol{x})$ is a function model. Since the generalization error is inaccessible because the true probability $p(y|\boldsymbol{x})p_{\mathrm{te}}(\boldsymbol{x})$ is unknown, empirical-error minimization is often used in practice (Vapnik, 1998):

$$\min_{f\in\mathcal{F}}\left[\frac{1}{n_{\mathrm{tr}}}\sum_{j=1}^{n_{\mathrm{tr}}}\mathrm{loss}(y_j^{\mathrm{tr}}, f(\boldsymbol{x}_j^{\mathrm{tr}}))\right].$$

However, under the covariate shift setup, plain empirical-error minimization is not *consistent* (i.e., it does not converge to the optimal function) if the model $\mathcal{F}$ is *misspecified* (i.e., the true function is not included in the model; see Shimodaira, 2000). Instead, the following *importance-weighted* empirical-error minimization is consistent under covariate shift:

$$\min_{f\in\mathcal{F}}\left[\frac{1}{n_{\mathrm{tr}}}\sum_{j=1}^{n_{\mathrm{tr}}}r(\boldsymbol{x}_j^{\mathrm{tr}})\mathrm{loss}(y_j^{\mathrm{tr}}, f(\boldsymbol{x}_j^{\mathrm{tr}}))\right],$$

where $r(\boldsymbol{x})$ is called the *importance* (Fishman, 1996) in the context of covariate shift adaptation:

$$r(\boldsymbol{x}) := \frac{p_{\mathrm{te}}(\boldsymbol{x})}{p_{\mathrm{tr}}(\boldsymbol{x})}.$$

However, since importance-weighted learning is not *statistically efficient* (i.e., it tends to have larger variance), slightly *flattening* the importance weights is practically useful for stabilizing the estimator. (Shimodaira, 2000) proposed to use the *exponentially-flattened importance weights* as

$$\min_{f \in \mathcal{F}} \left[\frac{1}{n_{\mathrm{tr}}} \sum_{j=1}^{n_{\mathrm{tr}}} r(\boldsymbol{x}_j^{\mathrm{tr}})^{\tau} \mathrm{loss}(y_j^{\mathrm{tr}}, f(\boldsymbol{x}_j^{\mathrm{tr}})) \right],$$

where $0 \leq \tau \leq 1$ is called the *exponential flattening parameter.* $\tau = 0$ corresponds to plain empirical-error minimization, while $\tau = 1$ corresponds to importance-weighted empirical-error minimization; $0 < \tau < 1$ will give an intermediate estimator that balances the trade-off between statistical efficiency and consistency. The exponential flattening parameter τ can be optimized by model selection criteria such as the *importance-weighted Akaike information criterion* for regular models (Shimodaira, 2000), the *importance-weighted subspace information criterion* for linear models (Sugiyama and Müller, 2005), and *importance-weighted cross-validation* for arbitrary models (Sugiyama et al., 2007).

One of the potential drawbacks of the above exponential flattering approach is that estimation of $r(\boldsymbol{x})$ (i.e., $\tau = 1$) is rather hard, as shown in this paper. Thus, when $r(\boldsymbol{x})$ is estimated poorly, all flattened weights $r(\boldsymbol{x})^{\tau}$ are also unreliable and then covariate shift adaptation does not work well in practice. To cope with this problem, we propose to use *relative importance weights* alternatively:

$$\min_{f \in \mathcal{F}} \left[\frac{1}{n_{\mathrm{tr}}} \sum_{j=1}^{n_{\mathrm{tr}}} r_{\alpha}(\boldsymbol{x}_j^{\mathrm{tr}}) \mathrm{loss}(y_j^{\mathrm{tr}}, f(\boldsymbol{x}_j^{\mathrm{tr}})) \right],$$

where $r_{\alpha}(\boldsymbol{x})$ $(0 \leq \alpha \leq 1)$ is the α-relative importance weight defined by

$$r_{\alpha}(\boldsymbol{x}) := \frac{p_{\mathrm{te}}(\boldsymbol{x})}{(1-\alpha)p_{\mathrm{te}}(\boldsymbol{x}) + \alpha p_{\mathrm{tr}}(\boldsymbol{x})}.$$

Note that, compared with the definition of the α-relative density-ratio (1), α and $(1-\alpha)$ are swapped in order to be consistent with exponential flattening. Indeed, the relative importance weights play a similar role to exponentially-flattened importance weights; $\alpha = 0$ corresponds to plain empirical-error minimization, while $\alpha = 1$ corresponds to importance-weighted empirical-error minimization; $0 < \alpha < 1$ will give an intermediate estimator that balances the trade-off between efficiency and consistency. We note that the relative importance weights and exponentially flattened importance weights agree only when $\alpha = \tau = 0$ and $\alpha = \tau = 1$; for $0 < \alpha = \tau < 1$, they are generally different.

A possible advantage of the above relative importance weights is that its estimation for $0 < \alpha < 1$ does not depend on that for $\alpha = 1$, unlike exponentially-flattened importance weights. Since α-relative importance weights for $0 < \alpha < 1$ can be reliably estimated by RuLSIF proposed in this paper, the performance of covariate shift adaptation is expected to be improved. Below, we experimentally investigate this effect.

4.3.2 Artificial Datasets

First, we illustrate how the proposed method behaves in covariate shift adaptation using one-dimensional artificial datasets.

In this experiment, we employ the following kernel regression model:

$$f(x;\boldsymbol{\beta}) = \sum_{i=1}^{n_{\mathrm{te}}} \beta_i \exp\left(-\frac{(x - x_i^{\mathrm{te}})^2}{2\rho^2}\right),$$

where $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_{n_{\mathrm{te}}})^\top$ is the parameter to be learned and ρ is the Gaussian width. The parameter $\boldsymbol{\beta}$ is learned by *relative importance-weighted least-squares* (RIW-LS):

$$\widehat{\boldsymbol{\beta}}_{\mathrm{RIW-LS}} = \operatorname*{argmin}_{\boldsymbol{\beta}} \left[\frac{1}{n_{\mathrm{tr}}} \sum_{j=1}^{n_{tr}} \widehat{r}_\alpha(x_j^{\mathrm{tr}}) \left(f(x_j^{\mathrm{tr}};\boldsymbol{\beta}) - y_j^{\mathrm{tr}}\right)^2\right],$$

or *exponentially-flattened importance-weighted least-squares* (EIW-LS):

$$\widehat{\boldsymbol{\beta}}_{\mathrm{EIW-LS}} = \operatorname*{argmin}_{\boldsymbol{\beta}} \left[\frac{1}{n_{\mathrm{tr}}} \sum_{j=1}^{n_{tr}} \widehat{r}(x_j^{\mathrm{tr}})^\tau \left(f(x_j^{\mathrm{tr}};\boldsymbol{\beta}) - y_j^{\mathrm{tr}}\right)^2\right].$$

The relative importance weight $\widehat{r}_\alpha(x_j^{\mathrm{tr}})$ is estimated by RuLSIF, and the exponentially-flattened importance weight $\widehat{r}(x_j^{\mathrm{tr}})^\tau$ is estimated by uLSIF (i.e., RuLSIF with $\alpha = 1$). The Gaussian width ρ is chosen by 5-fold *importance-weighted cross-validation* (Sugiyama et al., 2007).

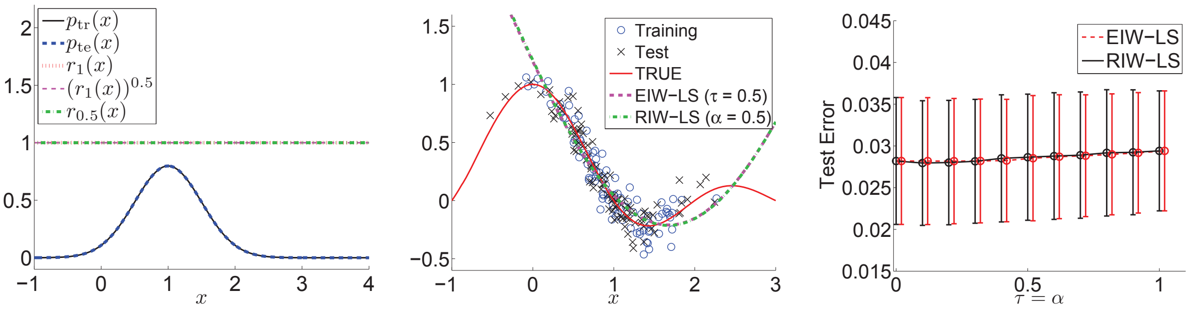
First, we consider the case where input distributions do not change:

$$P_{\mathrm{tr}} = P_{\mathrm{te}} = N(1, 0.25).$$

The densities and their ratios are plotted in Figure 6(a). The training output samples $\{y_j^{\mathrm{tr}}\}_{j=1}^{n_{\mathrm{tr}}}$ are generated as

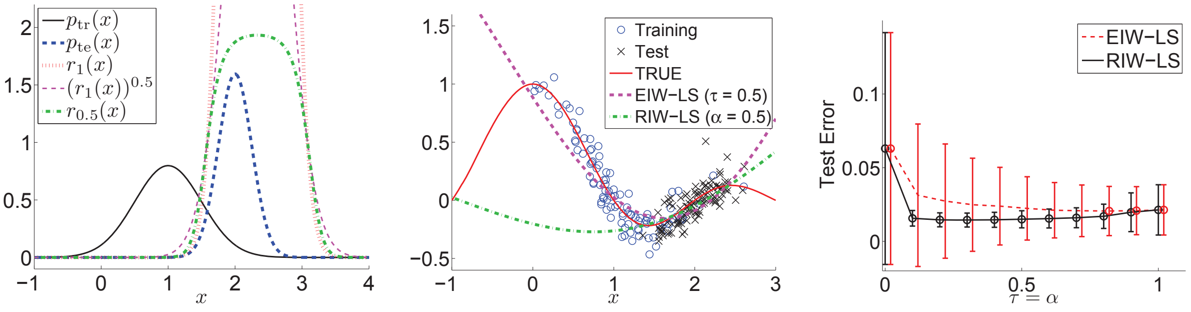
$$y_j^{\mathrm{tr}} = \mathrm{sinc}(x_j^{\mathrm{tr}}) + \epsilon_j^{\mathrm{tr}},$$

where $\{\epsilon_j^{\mathrm{tr}}\}_{j=1}^{n_{\mathrm{tr}}}$ is additive noise following $N(0, 0.01)$. We set $n_{\mathrm{tr}} = 100$ and $n_{\mathrm{te}} = 200$. Figure 6(b) shows a realization of training and test samples as well as learned functions obtained by RIW-LS with $\alpha = 0.5$ and EIW-LS with $\tau = 0.5$. This shows that RIW-LS with $\alpha = 0.5$ and EIW-LS with $\tau = 0.5$ give almost the same functions, and both functions fit the true function well in the test region. Figure 6(c) shows the mean and standard deviation of the test error under the squared loss over 200 runs, as functions of the relative flattening parameter α in RIW-LS and the exponential flattening parameter τ in EIW-LS. The method having a lower mean test error and another method that is comparable according to the *two-sample t-test* at the significance level 5% are specified by '$\circ$'. As can be observed, the proposed RIW-LS compares favorably with EIW-LS.



(a) Densities and ratios (b) Learned functions (c) Test error

Figure 6: Illustrative example of transfer learning under no distribution change.



(a) Densities and ratios (b) Learned functions (c) Test error

Figure 7: Illustrative example of transfer learning under covariate shift.

Next, we consider the situation where input distribution changes (Figure 7(a)):

$$P_{\mathrm{tr}} = N(1, 0.25),$$
$$P_{\mathrm{te}} = N(2, 0.1).$$

The output values are created in the same way as the previous case. Figure 7(b) shows a realization of training and test samples as well as learned functions obtained by RIW-LS with $\alpha = 0.5$ and EIW-LS with $\tau = 0.5$. This shows that RIW-LS with $\alpha = 0.5$ fits the true function slightly better than EIW-LS with $\tau = 0.5$ in the test region. Figure 7(c) shows that the proposed RIW-LS tends to outperform EIW-LS, and the standard deviation of the test error for RIW-LS is much smaller than EIW-LS. This is because EIW-LS with $0 < \tau < 1$ is based on an importance estimate with $\tau = 1$, which tends to have high fluctuation. Overall, the stabilization effect of relative importance estimation was shown to improve the test accuracy.

4.3.3 Real-World Datasets

Finally, we evaluate the proposed transfer learning method on a real-world transfer learning task.

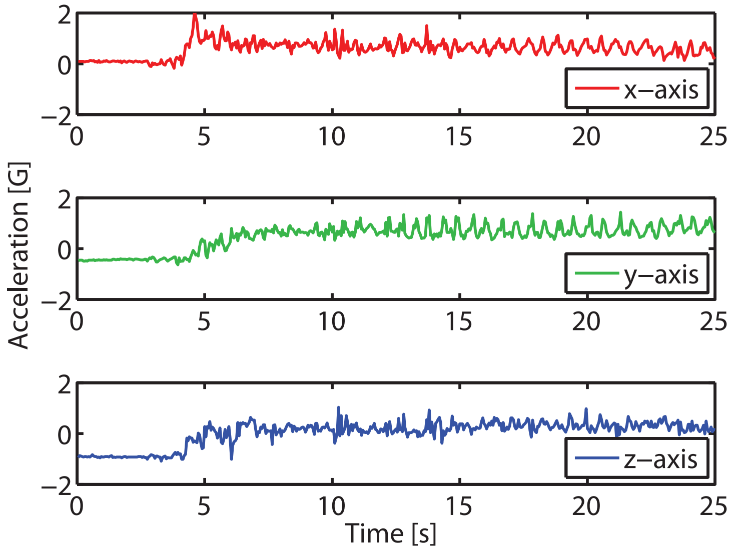


Figure 8: An example of three-axis accelerometer data for "walking" collected by *iPod touch*.

We consider the problem of human activity recognition from accelerometer data collected by *iPod touch*[3]. In the data collection procedure, subjects were asked to perform a specific action such as walking, running, and bicycle riding. The duration of each task was arbitrary and the sampling rate was 20Hz with small variations. An example of three-axis accelerometer data for "walking" is plotted in Figure 8.

To extract features from the accelerometer data, each data stream was segmented in a sliding window manner with window width 5 seconds and sliding step 1 second. Depending on subjects, the position and orientation of *iPod touch* was arbitrary—held by hand or kept in a pocket or a bag. For this reason, we decided to take the ℓ_2-norm of the 3-dimensional acceleration vector at each time step, and computed the following 5 orientation-invariant features from each window: *mean*, *standard deviation*, *fluctuation of amplitude*, *average energy*, and *frequency-domain entropy* (Bao and Intille, 2004; Bharatula et al., 2005).

Let us consider a situation where a new user wants to use the activity recognition system. However, since the new user is not willing to label his/her accelerometer data due to troublesomeness, no labeled sample is available for the new user. On the other hand, unlabeled samples for the new user and labeled data obtained from existing users are available. Let labeled training data $\{(\boldsymbol{x}_j^{\mathrm{tr}}, y_j^{\mathrm{tr}})\}_{j=1}^{n_{\mathrm{tr}}}$ be the set of labeled accelerometer data for 20 existing users. Each user has at most 100 labeled samples for each action. Let unlabeled test data $\{\boldsymbol{x}_i^{\mathrm{te}}\}_{i=1}^{n_{\mathrm{te}}}$ be unlabeled accelerometer data obtained from the new user.

We use *kernel logistic regression* (KLR) for activity recognition. We compare the following four methods:

[3] `http://alkan.mns.kyutech.ac.jp`

Table 5: Experimental results of transfer learning in real-world human activity recognition. Mean classification accuracy (and the standard deviation in the bracket) over 100 runs for activity recognition of a new user is reported. The method having the lowest mean classification accuracy and comparable methods according to the *two-sample t-test* at the significance level 5% are specified by bold face.

Task	KLR ($\alpha = 0$, $\tau = 0$)	RIW-KLR ($\alpha = 0.5$)	EIW-KLR ($\tau = 0.5$)	IW-KLR ($\alpha = 1$, $\tau = 1$)
Walks vs. run	0.803 (0.082)	**0.889(0.035)**	**0.882(0.039)**	**0.882 (0.035)**
Walks vs. bicycle	0.880 (0.025)	**0.892(0.035)**	0.867 (0.054)	0.854 (0.070)
Walks vs. train	0.985 (0.017)	**0.992(0.008)**	0.989 (0.011)	0.983 (0.021)

- Plain KLR without importance weights (i.e., $\alpha = 0$ or $\tau = 0$).
- KLR with relative importance weights for $\alpha = 0.5$.
- KLR with exponentially-flattened importance weights for $\tau = 0.5$.
- KLR with plain importance weights (i.e., $\alpha = 1$ or $\tau = 1$).

The experiments are repeated 100 times with different sample choice for $n_{\mathrm{tr}} = 500$ and $n_{\mathrm{te}} = 200$. Table 5 depicts the classification accuracy for three binary-classification tasks: *walk vs. run*, *walk vs. riding a bicycle*, and *walk vs. taking a train.* The classification accuracy is evaluated for 800 samples from the new user that are not used for classifier training (i.e., the 800 test samples are different from 200 unlabeled samples). The table shows that KLR with relative importance weights for $\alpha = 0.5$ compares favorably with other methods in terms of the classification accuracy. KLR with plain importance weights and KLR with exponentially-flattened importance weights for $\tau = 0.5$ are outperformed by KLR without importance weights in the *walk vs. riding a bicycle* task due to the instability of importance weight estimation for $\alpha = 1$ or $\tau = 1$.

Overall, the proposed relative density-ratio estimation method was shown to be useful also in transfer learning under covariate shift.

5 Conclusion

In this paper, we proposed to use a relative divergence for robust distribution comparison. We gave a computationally efficient method for estimating the relative Pearson divergence based on direct relative density-ratio approximation. We theoretically elucidated the convergence rate of the proposed divergence estimator under non-parametric setup, which showed that the proposed approach of estimating the relative Pearson divergence is more preferable than the existing approach of estimating the plain Pearson divergence. Furthermore, we proved that the asymptotic variance of the proposed divergence estimator is independent of the model complexity under a correctly-specified parametric setup. Thus, the proposed divergence estimator hardly overfits even with complex models. Experimentally, we demonstrated the practical usefulness of the proposed divergence estimator in two-sample homogeneity test, inlier-based outlier detection, and transductive transfer learning under covariate shift.

In addition to two-sample homogeneity test, outlier detection, and transfer learning, density ratios were shown to be useful for tackling various machine learning problems, including multi-task learning (Bickel et al., 2008; Simm et al., 2011), independence test (Sugiyama and Suzuki, 2011), feature selection (Suzuki et al., 2009), causal inference (Yamada and Sugiyama, 2010), independent component analysis (Suzuki and Sugiyama, 2011), dimensionality reduction (Suzuki and Sugiyama, 2010), unpaired data matching (Yamada and Sugiyama, 2011), clustering (Kimura and Sugiyama, 2011), conditional density estimation (Sugiyama et al., 2010), and probabilistic classification (Sugiyama, 2010). Thus, it would be promising to explore more applications of the proposed relative density-ratio approximator beyond two-sample homogeneity test, outlier detection, and transfer learning tasks.

Acknowledgments

MY was supported by the JST PRESTO program, TS was partially supported by MEXT KAKENHI 22700289 and Aihara Project, the FIRST program from JSPS, initiated by CSTP, TK was partially supported by Grant-in-Aid for Young Scientists (20700251), HH was supported by the FIRST program, and MS was partially supported by SCAT, AOARD, and the FIRST program.

A Technical Details of Non-Parametric Convergence Analysis

Here, we give the technical details of the non-parametric convergence analysis described in Section 3.1.

A.1 Results

For notational simplicity, we define linear operators $P, P_n, P', P'_{n'}$ as

$$\begin{aligned} Pf &:= \mathrm{E}_p f, \quad P_n f := \frac{\sum_{j=1}^{n} f(\boldsymbol{x}_j)}{n}, \\ P'f &:= \mathrm{E}_q f, \quad P'_{n'} f := \frac{\sum_{i=1}^{n'} f(\boldsymbol{x}'_i)}{n'}. \end{aligned}$$

For $\alpha \in [0,1]$, we define $S_{n,n'}$ and S as

$$S_{n,n'} = \alpha P_n + (1-\alpha)P'_{n'}, \quad S = \alpha P + (1-\alpha)P'.$$

We estimate the Pearson divergence between p and $\alpha p + (1-\alpha)q$ through estimating the density ratio

$$g^* := \frac{p}{\alpha p + (1-\alpha)p'}.$$

Let us consider the following density ratio estimator:

$$\begin{aligned}\widehat{g} :=& \mathop{\mathrm{argmin}}_{g\in\mathcal{G}} \left[\frac{1}{2}\left(\alpha P_n + (1-\alpha)P'_{n'}\right) g^2 - P_n g + \frac{\lambda_{\bar{n}}}{2}R(g)^2\right] \\ =& \mathop{\mathrm{argmin}}_{g\in\mathcal{G}} \left(\frac{1}{2}S_{n,n'}g^2 - P_n g + \frac{\lambda_{\bar{n}}}{2}R(g)^2\right).\end{aligned}$$

where $\bar{n} = \min(n, n')$ and $R(g)$ is a non-negative regularization functional such that

$$\sup_{\boldsymbol{x}}[|g(\boldsymbol{x})|] \leq R(g). \tag{17}$$

A possible estimator of the Pearson (PE) divergence $\widehat{\mathrm{PE}}_\alpha$ is

$$\widehat{\mathrm{PE}}_\alpha := P_n\widehat{g} - \frac{1}{2}S_{n,n'}\widehat{g}^2 - \frac{1}{2}.$$

Another possibility is

$$\widetilde{\mathrm{PE}}_\alpha := \frac{1}{2}P_n\widehat{g} - \frac{1}{2}.$$

A useful example is to use a *reproducing kernel Hilbert space* (RKHS; Aronszajn, 1950) as $\mathcal{G}$ and the RKHS norm as $R(g)$. Suppose $\mathcal{G}$ is an RKHS associated with bounded kernel $k(\cdot,\cdot)$:

$$\sup_{\boldsymbol{x}}[k(\boldsymbol{x},\boldsymbol{x})] \leq C.$$

Let $\|\cdot\|_{\mathcal{G}}$ denote the norm in the RKHS $\mathcal{G}$. Then $R(g) = \sqrt{C}\|g\|_{\mathcal{G}}$ satisfies Eq.(17):

$$g(\boldsymbol{x}) = \langle k(\boldsymbol{x},\cdot), g(\cdot)\rangle \leq \sqrt{k(\boldsymbol{x},\boldsymbol{x})}\|g\|_{\mathcal{G}} \leq \sqrt{C}\|g\|_{\mathcal{G}},$$

where we used the reproducing property of the kernel and Schwartz's inequality. Note that the Gaussian kernel satisfies this with $C = 1$. It is known that the Gaussian kernel RKHS spans a dense subset in the set of continuous functions. Another example of RKHSs is Sobolev space. The canonical norm for this space is the integral of the squared derivatives of functions. Thus the regularization term $R(g) = \|g\|_{\mathcal{G}}$ imposes the solution to be smooth. The RKHS technique in Sobolev space has been well exploited in the context of spline models (Wahba, 1990). We intend that the regularization term $R(g)$ is a generalization of the RKHS norm. Roughly speaking, $R(g)$ is like a "norm" of the function space $\mathcal{G}$.

We assume that the true density-ratio function $g^*(\boldsymbol{x})$ is contained in the model $\mathcal{G}$ and is bounded from above:

$$g^*(\boldsymbol{x}) \leq M_0 \quad \text{for all} \quad \boldsymbol{x} \in \mathcal{D}_{\mathrm{X}}.$$

Let $\mathcal{G}_M$ be a *ball* of $\mathcal{G}$ with radius $M > 0$:

$$\mathcal{G}_M := \{g \in \mathcal{G} \mid R(g) \leq M\}.$$

To derive the convergence rate of our estimator, we utilize the *bracketing entropy* that is a complexity measure of a function class (see p. 83 of van der Vaart and Wellner, 1996).

Definition 1 *Given two functions l and u, the bracket $[l, u]$ is the set of all functions f with $l(\boldsymbol{x}) \leq f(\boldsymbol{x}) \leq u(\boldsymbol{x})$ for all $\boldsymbol{x}$. An ϵ-bracket with respect to $L_2(\tilde{p})$ is a bracket $[l, u]$ with $\|l - u\|_{L_2(\tilde{p})} < \epsilon$. The bracketing entropy $\mathcal{H}_{[]}(\mathcal{F}, \epsilon, L_2(\tilde{p}))$ is the logarithm of the minimum number of ϵ-brackets with respect to $L_2(\tilde{p})$ needed to cover a function set $\mathcal{F}$.*

We assume that there exists γ $(0 < \gamma < 2)$ such that, for all $M > 0$,

$$\mathcal{H}_{[]}(\mathcal{G}_M, \epsilon, L_2(p)) = O\left(\left(\frac{M}{\epsilon}\right)^{\gamma}\right), \quad \mathcal{H}_{[]}(\mathcal{G}_M, \epsilon, L_2(p')) = O\left(\left(\frac{M}{\epsilon}\right)^{\gamma}\right). \tag{18}$$

This quantity represents a complexity of function class $\mathcal{G}$—the larger γ is, the more complex the function class $\mathcal{G}$ is because, for larger γ, more brackets are needed to cover the function class. The Gaussian RKHS satisfies this condition for arbitrarily small γ (Steinwart and Scovel, 2007). Note that when $R(g)$ is the RKHS norm, the condition (18) holds for all $M > 0$ if that holds for $M = 1$.

Then we have the following theorem.

Theorem 1 *Let $\bar{n} = \min(n, n')$, $M_0 = \|g^*\|_\infty$, and $c = (1+\alpha)\sqrt{P(g^* - Pg^*)^2} + (1-\alpha)\sqrt{P'(g^* - P'g^*)^2}$. Under the above setting, if $\lambda_{\bar{n}} \to 0$ and $\lambda_{\bar{n}}^{-1} = o(\bar{n}^{2/(2+\gamma)})$, then we have*

$$\widehat{\mathrm{PE}}_\alpha - \mathrm{PE}_\alpha = \mathcal{O}_p(\lambda_{\bar{n}} \max(1, R(g^*)^2) + \bar{n}^{-1/2} c M_0),$$

and

$$\widetilde{\mathrm{PE}}_\alpha - \mathrm{PE}_\alpha = \mathcal{O}_p(\lambda_{\bar{n}} \max\{1, M_0^{\frac{1}{2}(1-\frac{\gamma}{2})}, R(g^*) M_0^{\frac{1}{2}(1-\frac{\gamma}{2})}, R(g^*)\} + \lambda_{\bar{n}}^{\frac{1}{2}} \max\{M_0^{\frac{1}{2}}, M_0^{\frac{1}{2}} R(g^*)\}),$$

where $\mathcal{O}_p$ denotes the asymptotic order in probability.

In the proof of Theorem 1, we use the following auxiliary lemma.

Lemma 1 *Under the setting of Theorem 1, if $\lambda_{\bar{n}} \to 0$ and $\lambda_{\bar{n}}^{-1} = o(\bar{n}^{2/(2+\gamma)})$, then we have*

$$\|\widehat{g} - g^*\|_{L_2(S)} = \mathcal{O}_p(\lambda_{\bar{n}}^{1/2} \max\{1, R(g^*)\}), \quad R(\widehat{g}) = \mathcal{O}_p(\max\{1, R(g^*)\}),$$

where $\|\cdot\|_{L_2(S)}$ denotes the $L_2(\alpha p + (1-\alpha)q)$-norm.

A.2 Proof of Lemma 1

First, we prove Lemma 1.

From the definition, we obtain

$$\begin{aligned} &\frac{1}{2}S_{n,n'}\widehat{g}^2 - P_n\widehat{g} + \lambda_{\bar{n}}R(\widehat{g})^2 \leq \frac{1}{2}S_{n,n'}{g^*}^2 - P_ng^* + \lambda_{\bar{n}}R(g^*)^2 \\ \Rightarrow\quad &\frac{1}{2}S_{n,n'}(\widehat{g}-g^*)^2 - S_{n,n'}(g^*(g^*-\widehat{g})) - P_n(\widehat{g}-g^*) + \lambda_{\bar{n}}(R(\widehat{g})^2 - R(g^*)^2) \leq 0. \end{aligned}$$

On the other hand, $S(g^*(g^*-\widehat{g})) = P(g^*-\widehat{g})$ indicates

$$\begin{aligned} &\frac{1}{2}(S-S_{n,n'})(\widehat{g}-g^*)^2 - (S-S_{n,n'})(g^*(g^*-\widehat{g})) - (P-P_n)(\widehat{g}-g^*) - \lambda_{\bar{n}}(R(\widehat{g})^2 - R(g^*)^2) \\ &\geq \frac{1}{2}S(\widehat{g}-g^*)^2. \end{aligned}$$

Therefore, to bound $\|\widehat{g}-g^*\|_{L_2(S)}$, it suffices to bound the left-hand side of the above inequality.

Define $\mathcal{F}_M$ and $\mathcal{F}_M^2$ as

$$\mathcal{F}_M := \{g - g^* \mid g \in \mathcal{G}_M\} \quad \text{and} \quad \mathcal{F}_M^2 := \{f^2 \mid f \in \mathcal{F}_M\}.$$

To bound $|(S-S_{n,n'})(\widehat{g}-g^*)^2|$, we need to bound the bracketing entropies of $\mathcal{F}_M^2$. We show that

$$\begin{aligned} \mathcal{H}_{[]}(\mathcal{F}_M^2, \delta, L_2(p)) &= O\left(\left(\frac{(M+M_0)^2}{\delta}\right)^{\gamma}\right), \\ \mathcal{H}_{[]}(\mathcal{F}_M^2, \delta, L_2(q)) &= O\left(\left(\frac{(M+M_0)^2}{\delta}\right)^{\gamma}\right). \end{aligned}$$

This can be shown as follows. Let f_L and f_U be a δ-bracket for $\mathcal{G}_M$ with respect to $L_2(p)$; $f_L(x) \leq f_U(x)$ and $\|f_L - f_U\|_{L_2(p)} \leq \delta$. Without loss of generality, we can assume that $\|f_L\|_{L_\infty}, \|f_U\|_{L_\infty} \leq M + M_0$. Then f'_U and f'_L defined as

$$\begin{aligned} f'_U(x) &:= \max\{f_L^2(x), f_U^2(x)\}, \\ f'_L(x) &:= \begin{cases} \min\{f_L^2(x), f_U^2(x)\} & (\mathrm{sign}(f_L(x)) = \mathrm{sign}(f_U(x))), \\ 0 & (\text{otherwise}) \end{cases}, \end{aligned}$$

are also a bracket such that $f'_L \leq g^2 \leq f'_U$ for all $g \in \mathcal{G}_M$ s.t. $f_L \leq g \leq f_U$ and $\|f'_L - f'_U\|_{L_2(p)} \leq 2\delta(M+M_0)$ because $\|f_L - f_U\|_{L_2(p)} \leq \delta$ and the following relation is met:

$$
\begin{aligned}
(f'_L(x)-f'_U(x))^2 &\le \begin{cases}(f_L^2(x)-f_U^2(x))^2 & (\mathrm{sign}(f_L(x))=\mathrm{sign}(f_U(x))),\\ \max\{f_L^4(x),f_U^4(x)\} & (\text{otherwise})\end{cases}\\
&\le \begin{cases}(f_L(x)-f_U(x))^2(f_L(x)+f_U(x))^2 & (\mathrm{sign}(f_L(x))=\mathrm{sign}(f_U(x))),\\ \max\{f_L^4(x),f_U^4(x)\} & (\text{otherwise})\end{cases}\\
&\le \begin{cases}(f_L(x)-f_U(x))^2(f_L(x)+f_U(x))^2 & (\mathrm{sign}(f_L(x))=\mathrm{sign}(f_U(x))),\\ (f_L(x)-f_U(x))^2(|f_L(x)|+|f_U(x)|)^2 & (\text{otherwise})\end{cases}\\
&\le 4(f_L(x)-f_U(x))^2(M+M_0)^2.
\end{aligned}
$$

Therefore the condition for the bracketing entropies (18) gives $\mathcal{H}_{[]}(\mathcal{F}_M^2,\delta,L_2(p)) = O\left(\left(\frac{(M+M_0)^2}{\delta}\right)^{\gamma}\right)$. We can also show that $\mathcal{H}_{[]}(\mathcal{F}_M^2,\delta,L_2(q)) = O\left(\left(\frac{(M+M_0)^2}{\delta}\right)^{\gamma}\right)$ in the same fashion.

Let $f := \widehat{g}-g^*$. Then, as in Lemma 5.14 and Theorem 10.6 in (van de Geer, 2000), we obtain

$$
\begin{aligned}
|(S_{n,n'}-S)(f^2)| &\le \alpha|(P_n-P)(f^2)|+(1-\alpha)|(P'_{n'}-P')(f^2)|\\
=&\alpha\mathcal{O}_p\left(\frac{1}{\sqrt{\bar{n}}}\|f^2\|_{L_2(P)}^{1-\frac{\gamma}{2}}(1+R(\widehat{g})^2+M_0^2)^{\frac{\gamma}{2}}\vee \bar{n}^{-\frac{2}{2+\gamma}}(1+R(\widehat{g})^2+M_0^2)\right)\\
&+(1-\alpha)\mathcal{O}_p\left(\frac{1}{\sqrt{\bar{n}}}\|f^2\|_{L_2(P')}^{1-\frac{\gamma}{2}}(1+R(\widehat{g})^2+M_0^2)^{\frac{\gamma}{2}}\vee \bar{n}^{-\frac{2}{2+\gamma}}(1+R(\widehat{g})^2+M_0^2)\right)\\
\le&\mathcal{O}_p\left(\frac{1}{\sqrt{\bar{n}}}\|f^2\|_{L_2(S)}^{1-\frac{\gamma}{2}}(1+R(\widehat{g})^2+M_0^2)^{\frac{\gamma}{2}}\vee \bar{n}^{-\frac{2}{2+\gamma}}(1+R(\widehat{g})^2+M_0^2)\right),
\end{aligned}
\tag{19}
$$

where $a\vee b=\max(a,b)$ and we used

$$
\alpha\|f^2\|_{L_2(P)}^{1-\frac{\gamma}{2}}+(1-\alpha)\|f^2\|_{L_2(P')}^{1-\frac{\gamma}{2}} \le \left(\int f^4\mathrm{d}(\alpha P+(1-\alpha)P')\right)^{\frac{1}{2}(1-\frac{\gamma}{2})} = \|f^2\|_{L_2(S)}^{1-\frac{\gamma}{2}}
$$

by Jensen's inequality for a concave function. Since

$$
\|f^2\|_{L_2(S)} \le \|f\|_{L_2(S)}\sqrt{2(1+R(\widehat{g})^2+M_0^2)},
$$

the right-hand side of Eq.(19) is further bounded by

$$
\begin{aligned}
&|(S_{n,n'}-S)(f^2)|\\
&=\mathcal{O}_p\left(\frac{1}{\sqrt{\bar{n}}}\|f\|_{L_2(S)}^{1-\frac{\gamma}{2}}(1+R(\widehat{g})^2+M_0^2)^{\frac{1}{2}+\frac{\gamma}{4}}\vee \bar{n}^{-\frac{2}{2+\gamma}}(1+R(\widehat{g})^2+M_0^2)\right).
\end{aligned}
\tag{20}
$$

Similarly, we can show that

$$
\begin{aligned}
&|(S_{n,n'}-S)(g^*(g^*-\widehat{g}))|\\
&=\mathcal{O}_p\left(\frac{1}{\sqrt{\bar{n}}}\|f\|_{L_2(S)}^{1-\frac{\gamma}{2}}(1+R(\widehat{g})M_0+M_0^2)^{\frac{\gamma}{2}}\vee \bar{n}^{-\frac{2}{2+\gamma}}(1+R(\widehat{g})M_0+M_0^2)\right),
\end{aligned}
\tag{21}
$$

and

$$
\begin{aligned}
|(P_n - P)(g^* - \widehat{g})| &= \mathcal{O}_p\left(\frac{1}{\sqrt{\bar{n}}}\|f\|_{L_2(P)}^{1-\frac{\gamma}{2}}(1 + R(\widehat{g}) + M_0)^{\frac{\gamma}{2}} \vee \bar{n}^{-\frac{2}{2+\gamma}}(1 + R(\widehat{g}) + M_0)\right) \\
&\leq \mathcal{O}_p\left(\frac{1}{\sqrt{\bar{n}}}\|f\|_{L_2(S)}^{1-\frac{\gamma}{2}}(1 + R(\widehat{g}) + M_0)^{\frac{\gamma}{2}} M_0^{\frac{1}{2}(1-\frac{\gamma}{2})} \vee \bar{n}^{-\frac{2}{2+\gamma}}(1 + R(\widehat{g}) + M_0)\right), \qquad (22)
\end{aligned}
$$

where we used

$$
\|f\|_{L_2(P)} = \sqrt{\int f^2 \mathrm{d}P} = \sqrt{\int f^2 g^* \mathrm{d}S} \leq M_0^{\frac{1}{2}} \sqrt{\int f^2 \mathrm{d}S}
$$

in the last inequality. Combining Eqs.(20), (21), and (22), we can bound the $L_2(S)$-norm of f as

$$
\begin{aligned}
&\frac{1}{2}\|f\|_{L_2(S)}^2 + \lambda_{\bar{n}} R(\widehat{g})^2 \\
&\leq \lambda_{\bar{n}} R(g^*)^2 + \mathcal{O}_p\left(\frac{1}{\sqrt{\bar{n}}}\|f\|_{L_2(S)}^{1-\frac{\gamma}{2}}(1 + R(\widehat{g})^2 + M_0^2)^{\frac{1}{2}+\frac{\gamma}{4}} \vee \bar{n}^{-\frac{2}{2+\gamma}}(1 + R(\widehat{g})^2 + M_0^2)\right). \qquad (23)
\end{aligned}
$$

The following is similar to the argument in Theorem 10.6 in (van de Geer, 2000), but we give a simpler proof.

By Young's inequality, we have $a^{\frac{1}{2}-\frac{\gamma}{4}} b^{\frac{1}{2}+\frac{\gamma}{4}} \leq (\frac{1}{2} - \frac{\gamma}{4})a + (\frac{1}{2} + \frac{\gamma}{4})b \leq a + b$ for all $a, b > 0$. Applying this relation to Eq.(23), we obtain

$$
\begin{aligned}
&\frac{1}{2}\|f\|_{L_2(S)}^2 + \lambda_{\bar{n}} R(\widehat{g})^2 \\
&\leq \lambda_{\bar{n}} R(g^*)^2 + \mathcal{O}_p\left(\|f\|_{L_2(S)}^{2(\frac{1}{2}-\frac{\gamma}{4})}\left\{\bar{n}^{-\frac{2}{2+\gamma}}(1 + R(\widehat{g})^2 + M_0^2)\right\}^{\frac{1}{2}+\frac{\gamma}{4}} \vee \bar{n}^{-\frac{2}{2+\gamma}}(1 + R(\widehat{g})^2 + M_0^2)\right) \\
&\overset{\text{Young}}{\leq} \lambda_{\bar{n}} R(g^*)^2 + \frac{1}{4}\|f\|_{L_2(S)}^2 + \mathcal{O}_p\left(\bar{n}^{-\frac{2}{2+\gamma}}(1 + R(\widehat{g})^2 + M_0^2) + \bar{n}^{-\frac{2}{2+\gamma}}(1 + R(\widehat{g})^2 + M_0^2)\right) \\
&= \lambda_{\bar{n}} R(g^*)^2 + \frac{1}{4}\|f\|_{L_2(S)}^2 + \mathcal{O}_p\left(\bar{n}^{-\frac{2}{2+\gamma}}(1 + R(\widehat{g})^2 + M_0^2)\right),
\end{aligned}
$$

which indicates

$$
\frac{1}{4}\|f\|_{L_2(S)}^2 + \lambda_{\bar{n}} R(\widehat{g})^2 \leq \lambda_{\bar{n}} R(g^*)^2 + o_p\left(\lambda_{\bar{n}}(1 + R(\widehat{g})^2 + M_0^2)\right).
$$

Therefore, by moving $o_p(\lambda_{\bar{n}} R(\widehat{g})^2)$ to the left hind side, we obtain

$$
\begin{aligned}
\frac{1}{4}\|f\|_{L_2(S)}^2 + \lambda_{\bar{n}}(1 - o_p(1))R(\widehat{g})^2 &\leq \mathcal{O}_p\left(\lambda_{\bar{n}}(1 + R(g^*)^2 + M_0^2)\right) \\
&\leq \mathcal{O}_p\left(\lambda_{\bar{n}}(1 + R(g^*)^2)\right).
\end{aligned}
$$

This gives

$$
\begin{aligned}
&\|f\|_{L_2(S)} = \mathcal{O}_p(\lambda_{\bar{n}}^{\frac{1}{2}} \max\{1, R(g^*)\}), \\
&R(\widehat{g}) = \mathcal{O}_p(\sqrt{1 + R(g^*)^2}) = \mathcal{O}_p(\max\{1, R(g^*)\}).
\end{aligned}
$$

Consequently, the proof of Lemma 1 was completed.

A.3 Proof of Theorem 1

Based on Lemma 1, we prove Theorem 1.

As in the proof of Lemma 1, let $f := \widehat{g} - g^*$. Since $(\alpha P + (1-\alpha)P')(fg^*) = S(fg^*) = Pf$, we have

$$\begin{aligned}
\widehat{\mathrm{PE}}_\alpha - \mathrm{PE}_\alpha &= \frac{1}{2}S_{n,n'}\widehat{g}^2 - P_n\widehat{g} - (\frac{1}{2}Sg^{*2} - Pg^*) \\
&= \frac{1}{2}S_{n,n'}(f+g^*)^2 - P_n(f+g^*) - \left(\frac{1}{2}Sg^{*2} - Pg^*\right) \\
&= \frac{1}{2}Sf^2 + \frac{1}{2}(S_{n,n'} - S)f^2 + (S_{n,n'} - S)(g^*f) - (P_n - P)f \\
&\quad + \frac{1}{2}(S_{n,n'} - S)g^{*2} - (P_ng^* - Pg^*). \qquad (24)
\end{aligned}$$

Below, we show that each term of the right-hand side of the above equation is $\mathcal{O}_p(\lambda_{\bar{n}})$. By the central limit theorem, we have

$$\begin{aligned}
&\frac{1}{2}(S_{n,n'} - S)g^{*2} - (P_ng^* - Pg^*) \\
&\quad = \mathcal{O}_p\left(\bar{n}^{-1/2}M_0\left((1+\alpha)\sqrt{P(g^* - Pg^*)^2} + (1-\alpha)\sqrt{P'(g^* - P'g^*)^2}\right)\right).
\end{aligned}$$

Since Lemma 1 gives $\|f\|_2 = \mathcal{O}_p(\lambda_{\bar{n}}^{\frac{1}{2}}\max(1, R(g^*)))$ and $R(\widehat{g}) = \mathcal{O}_p(\max(1, R(g^*)))$, Eqs.(20), (21), and (22) in the proof of Lemma 1 imply

$$\begin{aligned}
|(S_{n,n'} - S)f^2| &= \mathcal{O}_p\left(\frac{1}{\sqrt{\bar{n}}}\|f\|_{L_2(S)}^{1-\frac{\gamma}{2}}(1 + R(g^*))^{1+\frac{\gamma}{2}} \vee \bar{n}^{-\frac{2}{2+\gamma}}R(g^*)^2\right) \\
&\leq \mathcal{O}_p(\lambda_{\bar{n}}\max(1, R(g^*)^2)), \\
|(S_{n,n'} - S)(g^*f)| &= \mathcal{O}_p\left(\frac{1}{\sqrt{\bar{n}}}\|f\|_{L_2(S)}^{1-\frac{\gamma}{2}}(1 + R(\widehat{g})M_0 + M_0^2)^{\frac{\gamma}{2}} \vee \bar{n}^{-\frac{2}{2+\gamma}}(1 + R(\widehat{g})M_0 + M_0^2)\right) \\
&\leq \mathcal{O}_p(\lambda_{\bar{n}}\max(1, R(g^*)M_0^{\frac{\gamma}{2}}, M_0^{\gamma}R(g^*)^{1-\frac{\gamma}{2}}, M_0R(g^*), M_0^2)) \\
&\leq \mathcal{O}_p(\lambda_{\bar{n}}\max(1, R(g^*)M_0^{\frac{\gamma}{2}}, M_0R(g^*))), \\
&\leq \mathcal{O}_p(\lambda_{\bar{n}}\max(1, R(g^*)^2)), \\
|(P_n - P)f| &\leq \mathcal{O}_p\left(\frac{1}{\sqrt{\bar{n}}}\|f\|_{L_2(S)}^{1-\frac{\gamma}{2}}(1 + R(\widehat{g}) + M_0)^{\frac{\gamma}{2}}M_0^{\frac{1}{2}(1-\frac{\gamma}{2})} \vee \bar{n}^{-\frac{2}{2+\gamma}}(1 + R(\widehat{g}) + M_0)\right) \\
&= \mathcal{O}_p(\lambda_{\bar{n}}\max(1, M_0^{\frac{1}{2}(1-\frac{\gamma}{2})}, R(g^*)M_0^{\frac{1}{2}(1-\frac{\gamma}{2})}, R(g^*))) \qquad (25) \\
&\leq \mathcal{O}_p(\lambda_{\bar{n}}\max(1, R(g^*)^2)),
\end{aligned}$$

where we used $\lambda_{\bar{n}}^{-1} = o(\bar{n}^{2/(2+\gamma)})$ and $M_0 \leq R(g^*)$. Lemma 1 also implies

$$Sf^2 = \|f\|_2^2 = \mathcal{O}_p(\lambda_{\bar{n}}\max(1, R(g^*)^2)).$$

Combining these inequalities with Eq.(24) implies

$$\widehat{\mathrm{PE}}_\alpha - \mathrm{PE}_\alpha = \mathcal{O}_p(\lambda_{\bar{n}} \max(1, R(g^*)^2) + n^{-1/2} c M_0),$$

where we again used $M_0 \leq R(g^*)$.

On the other hand, we have

$$\begin{aligned}\widetilde{\mathrm{PE}}_\alpha - \mathrm{PE}_\alpha &= \frac{1}{2} P_n \widehat{g} - \frac{1}{2} P g^* \\ &= \frac{1}{2} \left[(P_n - P)(\widehat{g} - g^*) + P(\widehat{g} - g^*) + (P_n - P) g^* \right]. \end{aligned} \tag{26}$$

Eq.(25) gives

$$(P_n - P)(\widehat{g} - g^*) = \mathcal{O}_p(\lambda_{\bar{n}} \max(1, M_0^{\frac{1}{2}(1-\frac{\gamma}{2})}, R(g^*) M_0^{\frac{1}{2}(1-\frac{\gamma}{2})}, R(g^*))).$$

We also have

$$P(\widehat{g} - g^*) \leq \|\widehat{g} - g^*\|_{L_2(P)} \leq \|\widehat{g} - g^*\|_{L_2(S)} M_0^{\frac{1}{2}} = \mathcal{O}_p(\lambda_{\bar{n}}^{\frac{1}{2}} \max(M_0^{\frac{1}{2}}, M_0^{\frac{1}{2}} R(g^*))),$$

and

$$(P_n - P) g^* = O_p(\bar{n}^{-\frac{1}{2}} \sqrt{P(g^* - P g^*)^2}) \leq O_p(\bar{n}^{-\frac{1}{2}} M_0) \leq \mathcal{O}_p(\lambda_{\bar{n}}^{\frac{1}{2}} \max(M_0^{\frac{1}{2}}, M_0^{\frac{1}{2}} R(g^*))),$$

Therefore by substituting these bounds into the relation (26), one observes that

$$\begin{aligned}&\widetilde{\mathrm{PE}}_\alpha - \mathrm{PE}_\alpha \\ =&\mathcal{O}_p(\lambda_{\bar{n}}^{\frac{1}{2}} \max(M_0^{\frac{1}{2}}, M_0^{\frac{1}{2}} R(g^*)) + \lambda_{\bar{n}} \max(1, M_0^{\frac{1}{2}(1-\frac{\gamma}{2})}, R(g^*) M_0^{\frac{1}{2}(1-\frac{\gamma}{2})}, R(g^*))).\end{aligned} \tag{27}$$

This completes the proof. ■

B Technical Details of Parametric Variance Analysis

Here, we give the technical details of the parametric variance analysis described in Section 3.2.

B.1 Results

For the estimation of the α-relative density-ratio (1), the statistical model

$$\mathcal{G} = \{ g(\boldsymbol{x}; \boldsymbol{\theta}) \mid \boldsymbol{\theta} \in \Theta \subset \mathbb{R}^b \}$$

is used where b is a finite number. Let us consider the following estimator of α-relative density-ratio,

$$\widehat{g} = \underset{g \in \mathcal{G}}{\operatorname{argmin}} \ \frac{1}{2} \left\{ \frac{\alpha}{n} \sum_{i=1}^{n} (g(\boldsymbol{x}_i))^2 + \frac{1-\alpha}{n'} \sum_{j=1}^{n'} (g(\boldsymbol{x}'_j))^2 \right\} - \frac{1}{n} \sum_{i=1}^{n} g(\boldsymbol{x}_i).$$

Suppose that the model is correctly specified, i.e., there exists $\boldsymbol{\theta}^*$ such that

$$g(\boldsymbol{x};\boldsymbol{\theta}^*) = r_\alpha(\boldsymbol{x}).$$

Then, under a mild assumption (see Theorem 5.23 of van der Vaart, 2000), the estimator $\widehat{g}$ is consistent and the estimated parameter $\widehat{\boldsymbol{\theta}}$ satisfies the asymptotic normality in the large sample limit. Then, a possible estimator of the α-relative Pearson divergence PE_α is

$$\widehat{\mathrm{PE}}_\alpha = \frac{1}{n}\sum_{i=1}^{n}\widehat{g}(\boldsymbol{x}_i) - \frac{1}{2}\left\{\frac{\alpha}{n}\sum_{i=1}^{n}(\widehat{g}(\boldsymbol{x}_i))^2 + \frac{1-\alpha}{n'}\sum_{j=1}^{n'}(\widehat{g}(\boldsymbol{x}'_j))^2\right\} - \frac{1}{2}.$$

Note that there are other possible estimators for PE_α such as

$$\widetilde{\mathrm{PE}}_\alpha = \frac{1}{2n}\sum_{i=1}^{n}\widehat{g}(\boldsymbol{x}_i) - \frac{1}{2}.$$

We study the asymptotic properties of $\widehat{\mathrm{PE}}_\alpha$. The expectation under the probability p (p') is denoted as $\mathbb{E}_{p(\boldsymbol{x})}[\cdot]$ ($\mathbb{E}_{p'(\boldsymbol{x})}[\cdot]$). Likewise, the variance is denoted as $\mathbb{V}_{p(\boldsymbol{x})}[\cdot]$ ($\mathbb{V}_{p'(\boldsymbol{x})}[\cdot]$). Then, we have the following theorem.

Theorem 2 *Let $\|r\|_\infty$ be the sup-norm of the standard density ratio $r(\boldsymbol{x})$, and $\|r_\alpha\|_\infty$ be the sup-norm of the α-relative density ratio, i.e.,*

$$\|r_\alpha\|_\infty = \frac{\|r\|_\infty}{\alpha\|r\|_\infty + 1 - \alpha}.$$

The variance of $\widehat{\mathrm{PE}}_\alpha$ is denoted as $\mathbb{V}[\widehat{\mathrm{PE}}_\alpha]$. Then, under the regularity condition for the asymptotic normality, we have the following upper bound of $\mathbb{V}[\widehat{\mathrm{PE}}_\alpha]$:

$$\begin{aligned}\mathbb{V}[\widehat{\mathrm{PE}}_\alpha] &= \frac{1}{n}\mathbb{V}_{p(\boldsymbol{x})}\left[r_\alpha - \frac{\alpha r_\alpha^2}{2}\right] + \frac{1}{n'}\mathbb{V}_{p'(\boldsymbol{x})}\left[\frac{(1-\alpha)r_\alpha^2}{2}\right] + o\left(\frac{1}{n},\frac{1}{n'}\right)\\ &\le \frac{\|r_\alpha\|_\infty^2}{n} + \frac{\alpha^2\|r_\alpha\|_\infty^4}{4n} + \frac{(1-\alpha)^2\|r_\alpha\|_\infty^4}{4n'} + o\left(\frac{1}{n},\frac{1}{n'}\right).\end{aligned}$$

Theorem 3 *The variance of $\widetilde{\mathrm{PE}}_\alpha$ is denoted as $\mathbb{V}[\widetilde{\mathrm{PE}}_\alpha]$. Let ∇g be the gradient vector of g with respect to $\boldsymbol{\theta}$ at $\boldsymbol{\theta} = \boldsymbol{\theta}^*$, i.e., $(\nabla g(\boldsymbol{x};\boldsymbol{\theta}^*))_j = \frac{\partial g(\boldsymbol{x};\boldsymbol{\theta}^*)}{\partial\theta_j}$. The matrix $\boldsymbol{U}_\alpha$ is defined by*

$$\boldsymbol{U}_\alpha = \alpha\mathbb{E}_{p(\boldsymbol{x})}[\nabla g\nabla g^\top] + (1-\alpha)\mathbb{E}_{p'(\boldsymbol{x})}[\nabla g\nabla g^\top].$$

Then, under the regularity condition, the variance of $\widetilde{\mathrm{PE}}_\alpha$ is asymptotically given as

$$\begin{aligned}\mathbb{V}[\widetilde{\mathrm{PE}}_\alpha] &= \frac{1}{n}\mathbb{V}_{p(\boldsymbol{x})}\left[\frac{r_\alpha + (1-\alpha r_\alpha)\mathbb{E}_{p(\boldsymbol{x})}[\nabla g]^\top\boldsymbol{U}_\alpha^{-1}\nabla g}{2}\right]\\ &\quad + \frac{1}{n'}\mathbb{V}_{p'(\boldsymbol{x})}\left[\frac{(1-\alpha)r_\alpha\mathbb{E}_{p(\boldsymbol{x})}[\nabla g]^\top\boldsymbol{U}_\alpha^{-1}\nabla g}{2}\right] + o\left(\frac{1}{n},\frac{1}{n'}\right).\end{aligned}$$

B.2 Proof of Theorem 2

Let $\widehat{\boldsymbol{\theta}}$ be the estimated parameter, i.e., $\widehat{g}(\boldsymbol{x}) = g(\boldsymbol{x}; \widehat{\boldsymbol{\theta}})$. Suppose that $r_\alpha(\boldsymbol{x}) = g(\boldsymbol{x}; \boldsymbol{\theta}^*) \in \mathcal{G}$ holds. Let $\delta\boldsymbol{\theta} = \widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}^*$, then the asymptotic expansion of $\widehat{\mathrm{PE}}_\alpha$ is given as

$$\begin{aligned}
\widehat{\mathrm{PE}}_\alpha &= \frac{1}{n}\sum_{i=1}^{n} g(\boldsymbol{x}_i; \widehat{\boldsymbol{\theta}}) - \frac{1}{2}\left\{\frac{\alpha}{n}\sum_{i=1}^{n} g(\boldsymbol{x}_i; \widehat{\boldsymbol{\theta}})^2 + \frac{1-\alpha}{n'}\sum_{j=1}^{n'} g(\boldsymbol{x}'_j; \widehat{\boldsymbol{\theta}})^2\right\} - \frac{1}{2} \\
&= \mathrm{PE}_\alpha + \frac{1}{n}\sum_{i=1}^{n}(r_\alpha(\boldsymbol{x}_i) - \mathbb{E}_{p(\boldsymbol{x})}[r_\alpha]) + \frac{1}{n}\sum_{i=1}^{n}\nabla g(\boldsymbol{x}_i; \boldsymbol{\theta}^*)^\top \delta\boldsymbol{\theta} \\
&\quad - \frac{1}{2}\left\{\frac{\alpha}{n}\sum_{i=1}^{n}(r_\alpha(\boldsymbol{x}_i)^2 - \mathbb{E}_{p(\boldsymbol{x})}[r_\alpha^2]) + \frac{1-\alpha}{n'}\sum_{j=1}^{n'}(r_\alpha(\boldsymbol{x}'_j)^2 - \mathbb{E}_{p'(\boldsymbol{x})}[r_\alpha^2])\right\} \\
&\quad - \left\{\frac{\alpha}{n}\sum_{i=1}^{n} r_\alpha(\boldsymbol{x}_i)\nabla g(\boldsymbol{x}_i; \boldsymbol{\theta}^*) + \frac{1-\alpha}{n'}\sum_{j=1}^{n'} r_\alpha(\boldsymbol{x}'_j)\nabla g(\boldsymbol{x}'_j; \boldsymbol{\theta}^*)\right\}^\top \delta\boldsymbol{\theta} + o_p\left(\frac{1}{\sqrt{n}}, \frac{1}{\sqrt{n'}}\right).
\end{aligned}$$

Let us define the linear operator G as

$$Gf = \frac{1}{\sqrt{n}}\sum_{i=1}^{n}(f(\boldsymbol{x}_i) - \mathbb{E}_{p(\boldsymbol{x})}[f]).$$

Likewise, the operator G' is defined for the samples from p'. Then, we have

$$\begin{aligned}
&\widehat{\mathrm{PE}}_\alpha - \mathrm{PE}_\alpha \\
&= \frac{1}{\sqrt{n}}G\big(r_\alpha - \frac{\alpha}{2}r_\alpha^2\big) - \frac{1}{\sqrt{n'}}G'\big(\frac{1-\alpha}{2}r_\alpha^2\big) \\
&\quad + \left\{\mathbb{E}_{p(\boldsymbol{x})}[\nabla g] - \alpha\mathbb{E}_{p(\boldsymbol{x})}[r_\alpha\nabla g] - (1-\alpha)\mathbb{E}_{p'(\boldsymbol{x})}[r_\alpha\nabla g]\right\}^\top \delta\boldsymbol{\theta} + o_p\left(\frac{1}{\sqrt{n}}, \frac{1}{\sqrt{n'}}\right) \\
&= \frac{1}{\sqrt{n}}G\big(r_\alpha - \frac{\alpha}{2}r_\alpha^2\big) - \frac{1}{\sqrt{n'}}G'\big(\frac{1-\alpha}{2}r_\alpha^2\big) + o_p\left(\frac{1}{\sqrt{n}}, \frac{1}{\sqrt{n'}}\right).
\end{aligned}$$

The second equality follows from

$$\mathbb{E}_{p(\boldsymbol{x})}[\nabla g] - \alpha\mathbb{E}_{p(\boldsymbol{x})}[r_\alpha\nabla g] - (1-\alpha)\mathbb{E}_{p'(\boldsymbol{x})}[r_\alpha\nabla g] = 0.$$

Then, the asymptotic variance is given as

$$\mathbb{V}[\widehat{\mathrm{PE}}_\alpha] = \frac{1}{n}\mathbb{V}_{p(\boldsymbol{x})}\left[r_\alpha - \frac{\alpha}{2}r_\alpha^2\right] + \frac{1}{n'}\mathbb{V}_{p'(\boldsymbol{x})}\left[\frac{1-\alpha}{2}r_\alpha^2\right] + o\left(\frac{1}{n}, \frac{1}{n'}\right). \tag{28}$$

We confirm that both $r_\alpha - \frac{\alpha}{2}r_\alpha^2$ and $\frac{1-\alpha}{2}r_\alpha^2$ are non-negative and increasing functions with respect to r for any $\alpha \in [0, 1]$. Since the result is trivial for $\alpha = 1$, we suppose $0 \le \alpha < 1$. The function $r_\alpha - \frac{\alpha}{2}r_\alpha^2$ is represented as

$$r_\alpha - \frac{\alpha}{2}r_\alpha^2 = \frac{r(\alpha r + 2 - 2\alpha)}{2(\alpha r + 1 - \alpha)^2},$$

and thus, we have $r_\alpha - \frac{\alpha}{2}r_\alpha^2 = 0$ for $r = 0$. In addition, the derivative is equal to

$$\frac{\partial}{\partial r}\frac{r(\alpha r + 2 - 2\alpha)}{2(\alpha r + 1 - \alpha)^2} = \frac{(1-\alpha)^2}{(\alpha r + 1 - \alpha)^3},$$

which is positive for $r \geq 0$ and $\alpha \in [0, 1)$. Hence, the function $r_\alpha - \frac{\alpha}{2}r_\alpha^2$ is non-negative and increasing with respect to r. Following the same line, we see that $\frac{1-\alpha}{2}r_\alpha^2$ is non-negative and increasing with respect to r. Thus, we have the following inequalities,

$$\begin{aligned}
&0 \leq r_\alpha(\boldsymbol{x}) - \frac{\alpha}{2}r_\alpha(\boldsymbol{x})^2 \leq \|r_\alpha\|_\infty - \frac{\alpha}{2}\|r_\alpha\|_\infty^2,\\
&0 \leq \frac{1-\alpha}{2}r_\alpha(\boldsymbol{x})^2 \leq \frac{1-\alpha}{2}\|r_\alpha\|_\infty^2.
\end{aligned}$$

As a result, upper bounds of the variances in Eq.(28) are given as

$$\begin{aligned}
\mathbb{V}_{p(\boldsymbol{x})}\left[r_\alpha - \frac{\alpha}{2}r_\alpha^2\right] &\leq \left(\|r_\alpha\|_\infty - \frac{\alpha}{2}\|r_\alpha\|_\infty^2\right)^2,\\
\mathbb{V}_{p'(\boldsymbol{x})}\left[\frac{1-\alpha}{2}r_\alpha^2\right] &\leq \frac{(1-\alpha)^2}{4}\|r_\alpha\|_\infty^4.
\end{aligned}$$

Therefore, the following inequality holds,

$$\begin{aligned}
\mathbb{V}[\widehat{\mathrm{PE}}_\alpha] &\leq \frac{1}{n}\left(\|r_\alpha\|_\infty - \frac{\alpha\|r_\alpha\|_\infty^2}{2}\right)^2 + \frac{1}{n'}\cdot\frac{(1-\alpha)^2\|r_\alpha\|_\infty^4}{4} + o\left(\frac{1}{n}, \frac{1}{n'}\right)\\
&\leq \frac{\|r_\alpha\|_\infty^2}{n} + \frac{\alpha^2\|r_\alpha\|_\infty^4}{4n} + \frac{(1-\alpha)^2\|r_\alpha\|_\infty^4}{4n'} + o\left(\frac{1}{n}, \frac{1}{n'}\right),
\end{aligned}$$

which completes the proof.

B.3 Proof of Theorem 3

The estimator $\widehat{\boldsymbol{\theta}}$ is the optimal solution of the following problem:

$$\min_{\theta\in\Theta}\left[\frac{1}{2n}\sum_{i=1}^{n}\alpha g(x_i;\boldsymbol{\theta})^2 + \frac{1}{2n'}\sum_{j=1}^{n'}(1-\alpha)g(x_j';\boldsymbol{\theta})^2 - \frac{1}{n}\sum_{i=1}^{n}g(x_i;\boldsymbol{\theta})\right].$$

Then, the extremal condition yields the equation,

$$\frac{\alpha}{n}\sum_{i=1}^{n}g(x_i;\widehat{\boldsymbol{\theta}})\nabla g(x_i;\widehat{\boldsymbol{\theta}}) + \frac{1-\alpha}{n'}\sum_{j=1}^{n'}g(x_j';\widehat{\boldsymbol{\theta}})\nabla g(x_j';\widehat{\boldsymbol{\theta}}) - \frac{1}{n}\sum_{i=1}^{n}\nabla g(x_i;\widehat{\boldsymbol{\theta}}) = 0.$$

Let $\delta\boldsymbol{\theta} = \widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}^*$. The asymptotic expansion of the above equation around $\boldsymbol{\theta} = \boldsymbol{\theta}^*$ leads to

$$\frac{1}{n}\sum_{i=1}^{n}(\alpha r_\alpha(x_i) - 1)\nabla g(x_i;\boldsymbol{\theta}^*) + \frac{1-\alpha}{n'}\sum_{j=1}^{n'}r_\alpha(x_j')\nabla g(x_j';\boldsymbol{\theta}^*) + \boldsymbol{U}_\alpha\delta\boldsymbol{\theta} + o_p\left(\frac{1}{\sqrt{n}}, \frac{1}{\sqrt{n'}}\right) = \boldsymbol{0}.$$

Therefore, we obtain

$$\delta\boldsymbol{\theta} = \frac{1}{\sqrt{n}}G((1-\alpha r_\alpha)\boldsymbol{U}_\alpha^{-1}\nabla g) - \frac{1}{\sqrt{n'}}G'((1-\alpha)r_\alpha\boldsymbol{U}_\alpha^{-1}\nabla g) + o_p\left(\frac{1}{\sqrt{n}}, \frac{1}{\sqrt{n'}}\right).$$

Next, we compute the asymptotic expansion of $\widetilde{\mathrm{PE}}_\alpha$:

$$\begin{aligned}\widetilde{\mathrm{PE}}_\alpha &= \frac{1}{2}\mathbb{E}_{p(\boldsymbol{x})}[r_\alpha] + \frac{1}{2n}\sum_{i=1}^{n}(r_\alpha(x_i) - \mathbb{E}_{p(\boldsymbol{x})}[r_\alpha]) \\ &\quad + \frac{1}{2n}\sum_{i=1}^{n}\nabla g(x_i;\boldsymbol{\theta}^*)^\top\delta\boldsymbol{\theta} - \frac{1}{2} + o_p\left(\frac{1}{\sqrt{n}}, \frac{1}{\sqrt{n'}}\right) \\ &= \mathrm{PE}_\alpha + \frac{1}{2\sqrt{n}}G(r_\alpha) + \frac{1}{2}\mathbb{E}_{p(\boldsymbol{x})}[\nabla g]^\top\delta\boldsymbol{\theta} + o_p\left(\frac{1}{\sqrt{n}}, \frac{1}{\sqrt{n'}}\right).\end{aligned}$$

Substituting $\delta\boldsymbol{\theta}$ into the above expansion, we have

$$\begin{aligned}\widetilde{\mathrm{PE}}_\alpha - \mathrm{PE}_\alpha &= \frac{1}{2\sqrt{n}}G(r_\alpha + (1-\alpha r_\alpha)\mathbb{E}_{p(\boldsymbol{x})}[\nabla g]^\top\boldsymbol{U}_\alpha^{-1}\nabla g) \\ &\quad - \frac{1}{2\sqrt{n'}}G'((1-\alpha)r_\alpha\mathbb{E}_{p(\boldsymbol{x})}[\nabla g]^\top\boldsymbol{U}_\alpha^{-1}\nabla g) + o_p\left(\frac{1}{\sqrt{n}}, \frac{1}{\sqrt{n'}}\right).\end{aligned}$$

As a result, we have

$$\begin{aligned}\mathbb{V}[\widetilde{\mathrm{PE}}_\alpha] &= \frac{1}{n}\mathbb{V}_{p(\boldsymbol{x})}\left[\frac{r_\alpha + (1-\alpha r_\alpha)\mathbb{E}_{p(\boldsymbol{x})}[\nabla g]^\top\boldsymbol{U}_\alpha^{-1}\nabla g}{2}\right] \\ &\quad + \frac{1}{n'}\mathbb{V}_{p'(\boldsymbol{x})}\left[\frac{(1-\alpha)r_\alpha\mathbb{E}_{p(\boldsymbol{x})}[\nabla g]^\top\boldsymbol{U}_\alpha^{-1}\nabla g}{2}\right] + o\left(\frac{1}{n}, \frac{1}{n'}\right),\end{aligned}$$

which completes the proof.

References

Ali, S. M. and Silvey, S. D. (1966). A general class of coefficients of divergence of one distribution from another. *Journal of the Royal Statistical Society, Series B*, 28:131–142.

Aronszajn, N. (1950). Theory of reproducing kernels. *Transactions of the American Mathematical Society*, 68:337–404.

Bao, L. and Intille, S. S. (2004). Activity recognition from user-annotated acceleration data. In *Proceedings of the 2nd IEEE International Conference on Pervasive Computing*, pages 1–17.

Bharatula, N. B., Stager, M., Lukowicz, P., and Troster, G. (2005). Empirical study of design choices in multi-sensor context recognition systems. In *Proceedings of the 2nd International Forum on Applied Wearable Computing*, pages 79–93.

Bickel, S., Bogojeska, J., Lengauer, T., and Scheffer, T. (2008). Multi-task learning for HIV therapy screening. In McCallum, A. and Roweis, S., editors, *Proceedings of 25th Annual International Conference on Machine Learning (ICML2008)*, pages 56–63.

Borgwardt, K. M., Gretton, A., Rasch, M. J., Kriegel, H.-P., Schölkopf, B., and Smola, A. J. (2006). Integrating structured biological data by kernel maximum mean discrepancy. *Bioinformatics*, 22(14):e49–e57.

Bradley, A. P. (1997). The use of the area under the ROC curve in the evaluation of machine learning algorithms. *Pattern Recognition*, 30:1145–1159.

Chang, C.-C. and Lin, C.-J. (2001). *LIBSVM: A Library for Support Vector Machines.* Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

Chapelle, O., Schölkopf, B., and Zien, A., editors (2006). *Semi-Supervised Learning.* MIT Press, Cambridge.

Cortes, C., Mansour, Y., and Mohri, M. (2010). Learning bounds for importance weighting. In Lafferty, J., Williams, C. K. I., Zemel, R., Shawe-Taylor, J., and Culotta, A., editors, *Advances in Neural Information Processing Systems 23*, pages 442–450.

Csiszár, I. (1967). Information-type measures of difference of probability distributions and indirect observation. *Studia Scientiarum Mathematicarum Hungarica*, 2:229–318.

Efron, B. and Tibshirani, R. J. (1993). *An Introduction to the Bootstrap.* Chapman & Hall, New York, NY.

Fishman, G. S. (1996). *Monte Carlo: Concepts, Algorithms, and Applications.* Springer-Verlag, Berlin.

Gretton, A., Borgwardt, K. M., Rasch, M., Schölkopf, B., and Smola, A. J. (2007). A kernel method for the two-sample-problem. In Schölkopf, B., Platt, J., and Hoffman, T., editors, *Advances in Neural Information Processing Systems 19*, pages 513–520. MIT Press, Cambridge, MA.

Hido, S., Tsuboi, Y., Kashima, H., Sugiyama, M., and Kanamori, T. (2011). Statistical outlier detection using direct density ratio estimation. *Knowledge and Information Systems*, 26(2):309–336.

Jiang, J. and Zhai, C. (2007). Instance weighting for domain adaptation in NLP. In *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*, pages 264–271.

Kain, A. and Macon, M. W. (1998). Spectral voice conversion for text-to-speech synthesis. In *Proceedings of 1998 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP1998)*, pages 285–288.

Kanamori, T., Hido, S., and Sugiyama, M. (2009). A least-squares approach to direct importance estimation. *Journal of Machine Learning Research*, 10:1391–1445.

Kimura, M. and Sugiyama, M. (2011). Dependence-maximization clustering with least-squares mutual information. *Journal of Advanced Computational Intelligence and Intelligent Informatics*.

Kullback, S. and Leibler, R. A. (1951). On information and sufficiency. *Annals of Mathematical Statistics*, 22:79–86.

Nguyen, X., Wainwright, M. J., and Jordan, M. I. (2010). Estimating divergence functionals and the likelihood ratio by convex risk minimization. *IEEE Transactions on Information Theory*, 56(11):5847–5861.

Pan, S. J. and Yang, Q. (2010). A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359.

Pearson, K. (1900). On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling. *Philosophical Magazine*, 50:157–175.

Rätsch, G., Onoda, T., and Müller, K.-R. (2001). Soft margins for adaboost. *Machine Learning*, 42(3):287–320.

Rockafellar, R. T. (1970). *Convex Analysis*. Princeton University Press, Princeton, NJ, USA.

Schölkopf, B., Platt, J. C., Shawe-Taylor, J., Smola, A. J., and Williamson, R. C. (2001). Estimating the support of a high-dimensional distribution. *Neural Computation*, 13(7):1443–1471.

Shimodaira, H. (2000). Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, 90(2):227–244.

Simm, J., Sugiyama, M., and Kato, T. (2011). Computationally efficient multi-task learning with least-squares probabilistic classifiers. *IPSJ Transactions on Computer Vision and Applications*, 3:1–8.

Smola, A., Song, L., and Teo, C. H. (2009). Relative novelty detection. In *Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics (AISTATS2009)*, pages 536–543.

Sriperumbudur, B., Fukumizu, K., Gretton, A., Lanckriet, G., and Schölkopf, B. (2009). Kernel choice and classifiability for RKHS embeddings of probability distributions. In Bengio, Y., Schuurmans, D., Lafferty, J., Williams, C. K. I., and Culotta, A., editors, *Advances in Neural Information Processing Systems 22*, pages 1750–1758. MIT Press, Cambridge, MA.

Steinwart, I. and Scovel, C. (2007). Fast rates for support vector machines using Gaussian kernels. *The Annals of Statistics*, 35(2):575–607.

Sugiyama, M. (2010). Superfast-trainable multi-class probabilistic classifier by least-squares posterior fitting. *IEICE Transactions on Information and Systems*, E93-D(10):2690–2701.

Sugiyama, M. and Kawanabe, M. (2012). *Covariate Shift Adaptation: Toward Machine Learning in Non-Stationary Environments*. MIT Press, Cambridge, MA, USA.

Sugiyama, M., Krauledat, M., and Müller, K.-R. (2007). Covariate shift adaptation by importance weighted cross validation. *Journal of Machine Learning Research*, 8:985–1005.

Sugiyama, M. and Müller, K.-R. (2005). Input-dependent estimation of generalization error under covariate shift. *Statistics & Decisions*, 23(4):249–279.

Sugiyama, M. and Suzuki, T. (2011). Least-squares independence test. *IEICE Transactions on Information and Systems*, E94-D(6):1333–1336.

Sugiyama, M., Suzuki, T., Itoh, Y., Kanamori, T., and Kimura, M. (2011). Least-squares two-sample test. *Neural Networks*, 24(7):735–751.

Sugiyama, M., Suzuki, T., Nakajima, S., Kashima, H., von Bünau, P., and Kawanabe, M. (2008). Direct importance estimation for covariate shift adaptation. *Annals of the Institute of Statistical Mathematics*, 60:699–746.

Sugiyama, M., Takeuchi, I., Suzuki, T., Kanamori, T., Hachiya, H., and Okanohara, D. (2010). Least-squares conditional density estimation. *IEICE Transactions on Information and Systems*, E93-D(3):583–594.

Suzuki, T. and Sugiyama, M. (2010). Sufficient dimension reduction via squared-loss mutual information estimation. In Teh, Y. W. and Tiggerington, M., editors, *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics (AISTATS2010)*, volume 9 of *JMLR Workshop and Conference Proceedings*, pages 804–811, Sardinia, Italy.

Suzuki, T. and Sugiyama, M. (2011). Least-squares independent component analysis. *Neural Computation*, 23(1):284–301.

Suzuki, T., Sugiyama, M., Kanamori, T., and Sese, J. (2009). Mutual information estimation reveals global associations between stimuli and biological processes. *BMC Bioinformatics*, 10(1):S52.

van de Geer, S. (2000). *Empirical Processes in M-Estimation*. Cambridge University Press.

van der Vaart, A. W. (2000). *Asymptotic Statistics*. Cambridge University Press.

van der Vaart, A. W. and Wellner, J. A. (1996). *Weak Convergence and Empirical Processes: With Applications to Statistics*. Springer, New York.

Vapnik, V. N. (1998). *Statistical Learning Theory*. Wiley, New York, NY.

Wahba, G. (1990). *Spline Model for Observational Data*. Society for Industrial and Applied Mathematics, Philadelphia and Pennsylvania.

Yamada, M. and Sugiyama, M. (2010). Dependence minimizing regression with model selection for non-linear causal inference under non-Gaussian noise. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence (AAAI2010)*, pages 643–648, Atlanta, Georgia, USA. The AAAI Press.

Yamada, M. and Sugiyama, M. (2011). Cross-domain object matching with model selection. In Gordon, G., Dunson, D., and Dudík, M., editors, *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics (AISTATS2011)*, Fort Lauderdale, Florida, USA.